

This Page Is Inserted by IFW Operations
and is not a part of the Official Record

BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images may include (but are not limited to):

- BLACK BORDERS
- TEXT CUT OFF AT TOP, BOTTOM OR SIDES
- FADED TEXT
- ILLEGIBLE TEXT
- SKEWED/SLANTED IMAGES
- COLORED PHOTOS
- BLACK OR VERY BLACK AND WHITE DARK PHOTOS
- GRAY SCALE DOCUMENTS

IMAGES ARE BEST AVAILABLE COPY.

**As rescanning documents *will not* correct images,
please do not report the images to the
Image Problem Mailbox.**

(19) 日本国特許庁 (JP)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開平 6-60121

(43) 公開日 平成6年(1994)3月4日

(51) Int. Cl.⁵
G 0 6 F 15/40
9/46
識別記号 庁内整理番号
5 1 0 Z 7218-5 L
5 0 0 Z 7218-5 L
3 4 0 C 8120-5 B
F I

技術表示箇所

審査請求 未請求 請求項の数 15

(全 33 頁)

(21) 出願番号 特願平5-58941

(22) 出願日 平成5年(1993)3月18日

(31) 優先権主張番号 特願平4-63064

(32) 優先日 平4(1992)3月19日

(33) 優先権主張国 日本 (JP)

(71) 出願人 000005108

株式会社日立製作所

東京都千代田区神田駿河台四丁目6番地

(72) 発明者 畠山 敦

東京都国分寺市東恋ヶ窪1丁目280番地 株式会社日立製作所中央研究所内

(72) 発明者 加藤 寛次

東京都国分寺市東恋ヶ窪1丁目280番地 株式会社日立製作所中央研究所内

(72) 発明者 浅川 悟志

神奈川県横浜市戸塚区戸塚町5030番地 株式会社日立製作所ソフトウェア開発本部内

(74) 代理人 弁理士 小川 勝男

最終頁に続く

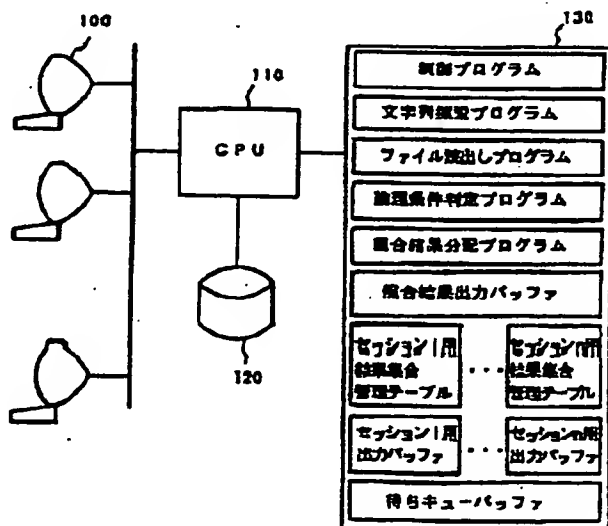
(54) 【発明の名称】 情報検索装置

(57) 【要約】 (修正有)

【目的】 文書検索システムを複数の端末を接続して使用する場合に、検索要求が同時に検索装置へ寄せられたときの、待ち時間を軽減する。

【構成】 検索処理中に受け付けた端末からの要求を蓄えておく待ちキューバッファ、及び複数の検索要求を一度に検索処理する文字列探索プログラムを備える。照合結果を要求端末毎に分配する分配プログラム及び要求元に対応してセッションn用出力バッファを設けることにより、同時に寄せられた検索要求を一括して処理する。また、要求元別に管理されたセッションn用結果集合管理テーブルを用いて、ハイラークサーチ時には、それぞれの要求元の結果集合のOR集合をベースとして検索し、結果を分配した後にそれぞれの要求元のベース集合とANDすることで、正しい結果を得ることができる。

図 8



1

【特許請求の範囲】

【請求項1】複数の検索端末が接続され、特定の文字列を含む文書を検索する情報検索装置において、検索処理中に他の検索要求を受け付けた場合、その検索要求を格納する手段と、前記検索要求格納手段内の検索要求が複数個格納されたとき、格納された複数の検索要求を一括して検索処理する手段とを設けたことを特徴とする情報検索装置。

【請求項2】請求項1記載の情報検索装置において、前記検索要求格納手段は、待ちキューバッファで構成されることを特徴とする情報検索装置。

【請求項3】請求項1記載の情報検索装置において、前記検索要求格納手段に、どの検索要求かを示す識別情報を格納し、文字列検索処理の終了後、前記識別情報に基づき検索結果を各検索要求元に振り分けて出力することを特徴とする情報検索装置。

【請求項4】請求項3記載の情報検索装置において、検索要求元毎に過去の検索結果を集合として蓄えておく結果集合管理格納手段を有し、前記結果集合に対してさらに条件を付加して、前記結果集合を絞り込むハイアラキ検索を行なうことを特徴とする情報検索装置。

【請求項5】請求項4記載の情報検索装置において、複数の検索要求を一括して検索処理する時には、まず全てのデータを対象に検索処理し、結果を検索要求元に振り分けた後、それぞれの検索要求元に対応する過去の結果集合と、新たな結果集合との間で、集合間AND処理を行い、絞り込み処理を実現することを特徴とする情報検索装置。

【請求項6】請求項4記載の情報検索装置において、複数の検索要求を一括して検索処理する時には、まずそれぞれの検索要求元に対応する結果集合のOR集合を基に検索処理し、結果を検索要求元別に振り分けた後、もう一度それぞれの前記OR集合を求めた検索要求元に対応する結果集合と、新たな結果集合との間で、集合間AND処理を行い、絞り込み処理を実現することを特徴とする情報検索装置。

【請求項7】請求項1記載の情報検索装置において、複数のデータベースを有し、前記検索要求を前記検索要求格納手段に格納するときに、どのデータベースを対象に検索するのかわかるデータベースidを付加し、複数の検索要求を一括して検索処理するときに、前記検索要求格納手段内のデータベースidを同じくする検索要求のみを選択的に前記検索要求格納手段から取り出して処理することを特徴とする情報検索装置。

【請求項8】複数の検索端末が接続され、かつそれぞれにデータを蓄えて特定の文字列を含む文書を検索する専用の文字列検索装置が複数個接続された情報検索装置において、各文字列検索装置が蓄えるデータの種別と文字列検索装置との対応表と、各文字列検索装置に送るべき検索要求を格納する検索要求格納手段を文字列検索装

2

置分設け、文字列検索装置の検索処理中に同じ文字列検索装置を使うような他の検索要求を受け付けた場合、当該文字列検索装置用の前記検索要求格納手段に検索要求を格納し、複数の検索要求が格納されたとき、複数の検索要求を一括して検索処理することを特徴とする情報検索装置。

【請求項9】請求項8記載の情報検索装置において、前記検索要求格納手段は、待ちキューバッファで構成されることを特徴とする情報検索装置。

10 【請求項10】請求項8記載の情報検索装置において、前記検索要求格納手段に、どの検索要求かを示す識別情報を格納し、文字列検索処理の終了後、前記識別情報に基づき検索結果を各検索要求元に振り分けて出力することを特徴とする情報検索装置。

20 【請求項11】請求項10記載の情報検索装置において、前記文字列検索装置は、検索要求元毎に過去の検索結果を集合として蓄えておく結果集合管理格納手段を有し、前記結果集合に対してさらに条件を付加して、前記結果集合を絞り込むハイアラキ検索を行なうことを特徴とする情報検索装置。

【請求項12】請求項11記載の情報検索装置において、前記情報検索装置にて複数の検索要求を一括して検索処理する時には、まず当該文字列検索装置が所有するデータの全てに対して検索処理し、結果を検索要求元別に振り分けた後、それぞれの検索要求元に対応する過去の結果集合と、新たな結果集合との間で、集合間AND処理を行い、その結果を前記情報検索装置に返すことによって絞り込み処理を実現することを特徴とする情報検索装置。

30 【請求項13】請求項11記載の情報検索装置において、前記情報検索装置にて複数の検索要求を一括して検索処理する時には、まずそれぞれの検索要求元に対応する結果集合のOR集合を基に検索処理し、結果を検索要求元別に振り分けた後、もう一度それぞれの前記OR集合を求めた検索要求元に対応する結果集合と、新たな結果集合との間で、集合間AND処理を行い、結果を前記情報検索装置に返すことによって絞り込み処理を実現することを特徴とする情報検索装置。

40 【請求項14】特定の文字列を含む文書を検索する情報検索装置において、複数件の検索要求を格納する手段と、前記検索要求格納手段内の検索要求が複数個格納されたとき、格納された複数の検索要求を一括して検索処理する手段とを設けたことを特徴とする情報検索装置。

50 【請求項15】特定の文字列を含む文書を検索する情報検索装置において、検索要求と要求元を示す識別情報に対応付けて複数個格納する手段と、前記検索要求格納手段内の検索要求が複数個格納されたとき、格納された複数の検索要求を一括して検索処理する手段と、検索処理の終了後前記識別情報に基づき検索結果を各検索要求元別に振り分けて出力する手段とを設けたことを特徴とす

る情報検索装置。

【発明の詳細な説明】

【0001】

【産業上の利用分野】本発明は、文書データベースを、文字列を指定して文書の全文を対象として探索する機能を複数のユーザが共用して使う情報検索装置に係り、特に各ユーザからの検索要求が同時に送られてくる場合に、待ち時間を最小限に抑えるのに好適な装置に関する。

【0002】

【従来の技術】従来より、文書の登録の際にキーワード付けを行う必要の無いフルテキストサーチ方式が特開平03-174652号公報で提案されている。この方式は、文書を単語単位に圧縮した凝縮本文と、使用文字を一文字単位で登録した文字成分表を用いて、フルテキストサーチを実用レベルで高速に行うことを目的としている。

【0003】しかしながら、凝縮されているとはいえ、テキストデータを一文字ずつ探索する方式であることに違いは無い。そのため、CPUがデータスキャンの間中文字の照合動作を行っており、他の処理を行う時間的余裕が無いことになる。このことは、複数のユーザにサービスを時分割に提供することが困難であるということを示す。

【0004】すなわち、フルテキストサーチ処理を行う検索装置に複数の端末が接続し、その各々の端末から頻繁に検索要求が与えられる場合、テキストスキャン中のため他の処理をCPUが行うことができず、CPUの文字照合動作が一通り終了するまで他の要求が待たれることになる。

【0005】また、処理の終了待ちとなっている状態で、次の処理要求が来た場合、あるいは、次々と各端末から検索要求が来た場合には、検索装置は要求が来た順番に処理をするために後から来た要求であるほど待ち時間が多くなるという問題があった。

【0006】

【発明が解決しようとする課題】本発明の目的は、複数の端末を持つフルテキストサーチの検索装置について、各端末からの検索要求が次々と送られてきても、見かけ上シングルユーザの場合と変わらない応答時間で端末に結果を帰すことのできる装置を提供することにある。

【0007】

【課題を解決するための手段】前記課題を解決するために、検索処理中に他の検索端末からの検索要求を受け付けた場合、その検索要求を格納する手段と、検索要求格納手段内の検索要求の個数を監視して、予め設定された個数n個以上の複数の検索要求が格納されたとき、格納された複数の検索要求を一括して検索処理する手段とを設けたことにある。

【0008】具体的には、以下の手段を有する装置を構

成する。

(1) 文字列の照合手段

(2) テキストデータの格納と読出し手段

(3) 複数の検索端末より要求を受け付ける手段

(4) 検索要求の待ちキューバッファ（前記検索要求格納手段に相当する）

(5) 検索要求元識別手段

(6) キューバッファに溜った検索要求を要求元の識別子を付加して一つにまとめる検索要求を統一する手段

10 (7) 要求元別に検索結果の文書集合を格納する手段

(8) 要求元別に結果情報を一時保存し、要求元へ転送する手段

(9) 統一された検索要求に対して一時に検索処理を行い、その結果を要求元別に振り分ける手段

以下これらの手段を用いた検索処理の概要を説明する。

【0009】検索処理中でも、他の端末からの要求を受け付けられるようにするには、図1のようにキューバッファを設け、検索中に受け付けた要求を逐次待ちキューとして登録していく方法が考えられる。しかし、この方法では、後から登録した検索要求はその前に登録してある検索要求の検索処理が全て実行されるまで待たされることになる。そこで、図2に示すように検索要求をバッファから取り出す時に、バッファに既に入っている検索要求の数をみて、溜っている検索要求を一つにまとめる検索要求統一処理を行うことで前記の検索要求登録順に係らず一回の検索処理で全ての要求元に検索結果を送出することができる。以下、この検索要求統一処理と要求元への検索結果送出手法について概略を説明する。

【0010】この検索要求統一処理は、例えば「「計算機」という単語が現われる文書を探せ」という要求と、
30 「「バイオ技術」という単語が現われる文書を探せ」という要求と「「学習型ユーザインタフェース」という単語が現われる文書を探せ」という要求がキューバッファに格納されていた場合に、これらをまとめて「「計算機」または「バイオ技術」または「学習型ユーザインタフェース」のいずれかの単語が現われる文書を探せ」という新たな要求として検索処理を実行する。

【0011】この検索要求を管理する待ちキューは、検索要求のあった端末の識別子を条件式と共に格納するものである。すなわち、前記例での検索要求は以下のように表せる。

【0012】最初の段階で

キュー1=u1: 「計算機」

キュー2=u2: 「バイオ技術」

キュー3=u3: 「学習型ユーザインタフェース」

とキューバッファに入っていた場合、これらの検索要求を統一して

キュー1=u1: 「計算機」 o r

u2: 「バイオ技術」 o r

u3: 「学習型ユーザインタフェース」

5

のように新しい検索要求として検索処理する。ここで、 $u1$, $u2$, $u3$ は要求元の識別子である。

【0013】このようにして統一された検索要求で、検索処理してすべての要求に対していずれかの条件式を充たす文書を検索する。すなわち、前記の例では、“計算機”、“パイオ技術”、“学習型ユーザインタフェース”のいずれかの語を含む文書を検索し、もしこれらの語を含む文書がヒットすれば、文書特定する文書IDと共に要求元の識別子を出力する。例えば、 $d..$ を文書IDとして、

$d1, u1, u3, u5$

$d3, u2$

$d10, u3, u4$

$d25, u2, u5$

$d37, u5$

のように出力する。この例では、例えば $d1$ の文書について、 $u1, u3, u5$ の要求した条件式が充たされていることを示す。

【0014】次にこの出力結果を要求元へ振り分ける処理を行う。この振り分け処理は、要求元識別子をもとにして行うことができる。すなわち、前記例では、 $u..$ によってそれぞれの要求元への検索結果集合として格納し各要求元へ出力する。

【0015】例えば、

$u1: (d1)$

$u2: (d3, d25)$

$u3: (d1, d10)$

$u4: (d10)$

$u5: (d1, d25, d37)$

のようになる。これは、例えば $u1$ の要求した条件に合致する文書が $d1$ であったことを示している。最後にこのようにして得られた要求元毎の検索結果を各要求元へ出力して処理を終える。

【0016】このようにして、複数の検索端末の要求をひとつにまとめて検索処理することにより、検索装置のレスポンスを向上することができる。例えば、前記例では、5個の検索要求を統一して検索処理しているので、本来5回テキストをスキャンしなければならないところを1回のスキャンで処理を終えることができる。

【0017】次に、検索装置にさらに高度な機能を持たせた場合の処理の概要について、説明する。フルテキストサーチにおいては、テキストをスキャンするというその方式上の問題から、できるだけスキャンする量を減らした方が結果として処理時間が短縮するという特徴がある。また、検索の手順を考えた場合、前回の検索結果集合をもとに、さらに条件を付加して絞り込みの検索をすることが圧倒的に多い。すなわち、一般的な検索では、既に絞られた特定の集合に対してキーワードを変えながらさらに絞り込んでいくことが多く、常にDB全体(原集合)に対しての検索を行うわけではない。従って、絞

6

り込みの検索を行うのであれば、すべてのデータをスキャンするよりも、前回の検索結果集合のデータのみについてスキャンする方が、処理時間を短縮することができる。この様な、前回の検索結果である限られた文書集合を次の検索対象として検索処理していき、検索処理毎にその探索範囲を狭めていく検索方法をここではハイアラキ検索と呼ぶ。

【0018】ハイアラキ検索機能を複数端末接続のシステムで実現するには、まず検索対象をどの様に管理し、複数要求の受け付け時どの様に設定するかが問題となる。この問題を図3を用いて具体的に説明する。図3では、時刻 $t1$ にユーザ a が“音声”という検索タームがある文書集合 $BASE_A1$ をつくり、時刻 $t3$ に $BASE_A1$ に対して、更に“合成”という検索タームで絞り込んでいる。一方、時刻 $t2$ にはユーザ B が“画像”という検索タームがある文書集合 $BASE_B1$ をつくり、時刻 $t4$ に $BASE_B1$ に対して、更に“認識”という検索タームで絞り込んでいる。それぞれのユーザの第2回目の検索では、 $BASE_A1$, $BASE_B1$ という異なる文書集合を対象に別々の検索処理を行う必要があり、複数個の検索要求を一括して処理することが困難となる。つまり、時刻 $t3$ と $t4$ での文字列検索処理を同時に行うことができない。これまで説明してきたように、文字列検索処理は非常に時間のかかる処理なので、より多くの時間を検索サービスに要するようになってしまう。

【0019】この問題を解決するには、別々の検索対象の文書集合を一つにまとめ、一度の検索処理で複数個の検索要求を処理する必要がある。これは例えば、図4に示すように常にデータベースの全体を検索対象として、検索処理を行い検索処理後に各要求元の検索対象との積集合を求めることで解決することができる。この方法では、常にDB全体(原集合) $BASE_0$ を対象に検索処理するので、ユーザ A , ユーザ B の2回目の検索要求も時刻 $t3$ の一度の文字列検索処理で終了することができる。すなわち、時刻 $t3$ において $BASE_0$ に対してユーザ A とユーザ B の検索要求である“合成”と“認識”のいずれかを含む文書を文字列検索し、検索処理終了後にユーザ A とユーザ B の検索結果集合に分配し、それぞれ $BASE_A1$, $BASE_B1$ と積集合を求めることで、求める検索結果集合 $BASE_A2$, $BASE_B2$ を得ることができる。この方法では、図3の $t3$ 及び $t4$ の文字列検索処理を一度に行うことができるが、常に全データベース $BASE_0$ のテキストデータをスキャンしなければならないため、探索量が増えて処理時間が多量にかかるという欠点がある。

【0020】これに対し、図5のように $BASE_A1$ と $BASE_B1$ の和集合を検索対象にして一括検索する方法も考えられる。確かに検索対象は少なくなるので探索量は減少するが、単純に $BASE_A1$ と $BASE$

BASE_B1の和集合BASE_AB1の検索結果をユーザAとユーザBに分配するだけでは、求める結果集合が得られない。すなわち、図に示す交差斜線の部分(ユーザAの例では、BASE_B1でBASE_A1以外であり、かつ“合成”を含む文書)がノイズとなる。このように、マルチユーザ環境で検索装置を運用する場合、検索対象DBが同一でも各ユーザの検索文字列がそれぞれ異なり、また絞り込み回数もそれぞれ異なるため、通常ある時点での各ユーザの検索対象集合は相互に違ったものになる。したがって、各ユーザの検索要求(検索文字列)を一括して検索しようとしても対象とする集合が異なるため、単にそれぞれの検索対象集合の論理和(OR)を取り、それに対して検索を行ったとしても、正しい検索結果集合が得られない。

【0021】これを解決するには、図6のように一括検索処理した後、それぞれの要求元の検索結果集合との積集合を求め、これを検索結果とする。すなわち、時刻t*

$$\begin{aligned} \text{BASE_AB3A} &= \text{BASE_A1 AND BASE_AB2A} \\ &= \text{BASE_A1 AND BASE_AB2(“合成”)} \\ &= \text{BASE_A1 AND BASE_AB1((“合成” OR “認識”) AND “合成”)} \\ &= \text{BASE_A1 AND BASE_AB1(“合成”)} \\ &= \text{BASE_A1 AND (BASE_A1(“合成”) OR BASE_B1(“合成”))} \\ &= (\text{BASE_A1 AND BASE_A1(“合成”)}) \\ &\quad \text{OR (BASE_A1 AND BASE_B1(“合成”))} \\ &= \text{BASE_A1(“合成”) OR (BASE_A1 AND BASE_B1(“合成”))} \end{aligned}$$

図7は以上の検索結果集合間の関係を示すものである。図に示すように、BASE_A1とBASE_B1の積集合(BASE_A1 AND BASE_B1)はBASE_A1に完全に包含される。従って、(BASE_A1 AND BASE_B1)(“合成”)も、BASE_A1(“合成”)に完全に包含される。ゆえに、
 $\text{BASE_AB3A} = \text{BASE_A1(“合成”)}$
 $= \text{BASE_A2}$
 が成り立つ。

【0024】このような、集合間の論理条件の演算は、文字列検索に較べ極めて短時間に処理できるため、文字列検索処理の対象をできるだけ減少させている本方式は有効で、複数人の検索要求を一括して短時間で処理することができるといえる。

【0025】このようなハイアラキー検索機能を有する検索装置において、前記の複数要求がキューバッファに溜っている場合の処理についてもう少し具体的に説明する。

【0026】例えば各検索端末の検索要求により、現在までに次のように結果集合が得られているものとする。

【0027】

u1: (d1, d5, d12, d15, d18, d27)

u2: (d2, d5, d7, d12, d18)

u3: (d1, d5, d18, d30)

u4: (d3, d7, d12, d30, d42, d5)

*3での検索結果集合BASE_AB2をユーザAとユーザBのそれぞれに分配すると共に、各ユーザの検索対象であるBASE_A1とBASE_B1との積集合を取るのである。こうすれば、要求元単位で、検索ノイズのない正しい検索結果が得られることになる。

【0022】以下、図6の方式でえられたユーザAの検索結果BASE_AB3Aが求める集合BASE_A2に等しいことを図7を用いて説明する。ここでは、検索対象を定めて文字列検索した結果をBASE_xx

(“文字列”)のように表すものとする。この式は、BASE_xxの集合を対象に“文字列”のある文書を検索した結果集合を表す。例えば、BASE_0(“音声”)はBASE_0を対象に“音声”で検索した結果集合なので、BASE_A1と同一集合を表すことになる。図6の検索結果集合BASE_AB3Aはこの式を用いて以下のように展開できる。

【0023】

0, d52)

u5: (d2, d5, d8, d12, d18, d42, d52)

この時、キューには次のように処理すべき要求が入っていたものとする。

【0028】キュー1=u1: “計算機”

キュー2=u2: “バイオ技術”

キュー3=u3: “学習型ユーザインタフェース”

検索要求のあるのはu1, u2, u3の端末からであるので、前記検索結果集合のうち、u1, u2, u3の結果集合のOR集合を求めて、この集合内の文書のテキストデータのみをスキャンして“計算機” “バイオ技術”

“学習型ユーザインタフェース”のいずれかの語を含む文書を探索して、ヒットした文書について、要求元の識別子と共に文書IDを出力する。すなわち、下記の結果集合

u1 or u2 or u3: (d1, d2, d5, d7, d12, d15, d18, d27, d30)

に含まれる文書のテキストデータをスキャンし、例えば次のような検索結果を得る。

【0029】d1, u1, u3

d5, u2

d12, u2, u3

d15, u1

d30, u2

この後、要求元別にこの出力結果を振り分けるが、ただ振り分けただけでは、要求元別の検索結果集合に含まれない文書が出力されるため、振り分けた後の集合と結果集合とのAND集合を求め最終結果とする。すなわち、一旦下記のように要求元別に検索結果を振り分けた後、

u1: (d1, d15)

u2: (d5, d12, d30)

u3: (d1, d12)

前記、最初にOR集合を求めた結果集合とのAND集合を求め、

u1: (d1, d15)

u2: (d5, d12)

u3: (d1)

とすると、正しい結果が得られることになる。

【0030】このように、検索結果集合をもとにスキャンすべき文書をあらかじめ絞っておき、検索要求をORで検索処理した後、各要求元別の結果集合をANDすることにより、レスポンスがよく、なおかつハイアラキ検索という機能を提供することが可能となる。

【0031】

【作用】以上の手段を用いて、複数の検索端末から検索要求をキューバッファに溜め込み、一定数以上の検索要求がバッファに入ったときには、検索要求を一つに統一して処理することにより、待ち時間の少ない検索装置を提供することができる。統一された検索要求には、要求元の識別子がそれぞれの検索要求に付加し、文字列の照合結果にも要求元識別子を出力するので、一度に検索処理した後、結果集合を要求元別に振り分けることができる。

【0032】また、要求元別に過去の検索結果集合を格納する手段を有し、絞り込み検索のモードの時、統一された検索要求に対しては、まず検索要求元毎の結果集合のOR集合を検索の対象として、テキストデータをスキャンすることにより、データスキャン量を少なくして、検索処理時間を短縮することができる。このとき、検索結果に対して、前記OR集合をとった結果集合ともう一度今度はANDを求めることで、各要求元別に正しい絞り込み結果が得ることができる。

【0033】

【実施例】以下、本発明の実施例について詳細に説明する。

【0034】図8は、本実施例の構成を示す図である。本実施例は、検索端末100、検索処理を実行するCPU110、データベースを格納する磁気ディスク120、各種のプログラムとデータバッファを格納するメモリ130からなっている。メモリ130には、検索処理全体を制御する制御プログラム、データベース中の特定文字列を探索する文字列探索プログラム、データを前記磁気ディスクから読み出し文字列探索プログラムにデータを送るファイル読み出しプログラム、文字列探索プロ

グラムの出力から検索文字列間のAND、ORを判定処理する論理条件判定プログラム、要求が複数の端末から送られたものである場合、各要求元へ検索結果を分配して送出する照合結果分配プログラムの各プログラムが格納してある。また、同メモリのほかの部分には文字列探索及び論理条件判定の結果を一時的に蓄える照合結果出力バッファ、各要求元別に過去の検索結果集合を管理蓄積するセッションn用出力バッファ、各要求元へ検索結果を出力するためのセッションn用出力バッファ、複数の端末から同時に要求が来たときに検索要求を一時格納しておく待ちキューバッファとしても使っている。

【0035】これより、検索処理の流れに従って、各プログラムの処理の詳細を説明する。まず、端末側での検索処理の流れについて説明する。図9は本実施例での検索端末で実行する業務の流れを示すPAD図である。本実施例では、検索端末を検索装置に接続する接続要求を行なった後、条件式を端末使用者から読み込み、検索要求を装置に送り結果を受け取る処理を繰り返すものとする。受け取った検索結果により、場合によっては文書データを検索装置より取得して表示することも行なう。すべての検索業務が終了した後、検索装置との接続を開放するものとする。

【0036】図10は制御プログラムの処理をまとめたPAD図である。

【0037】まず検索端末より接続要求があると、制御プログラムはセッションn用結果集合管理テーブルとセッションn用出力バッファを確保する。このnは、接続要求のあった端末ごとに更新される1からの通し番号である。この番号を以後端末識別子と呼ぶ。例えば、既に2個の端末からの接続要求が来ていて、新たに3個目の端末から接続要求が来た場合、この端末の識別子を3とし、セッション3用結果集合管理テーブル及びセッション3用出力バッファの領域を確保する。

【0038】接続の開放要求があった場合には、上記端末識別子の結果集合管理テーブルと出力バッファのメモリ領域を開放する。

【0039】文書データの表示要求の場合は、要求のあった文書をファイルから読みだす命令をファイル読み出しプログラムに対して送り、該プログラムがセッションn用出力バッファに出力する文書データを要求元へ送出する。

【0040】制御プログラムの処理の説明の最後に、本発明の特徴である検索要求が端末より送られたときの処理について詳細に説明する。

【0041】検索要求を受け付けた場合、制御プログラムは新たに待ちキューを生成してキューバッファに積み上げる処理を行なう。このキューバッファは図11に示すように、要求元の端末識別子と検索条件式を対にして積み上げた構造となっている。キューが一つ積み上がる毎にキューカウンタの数をインクリメントする。後に説

明する文字列探索プログラムがこのキューバッファから検索要求を取りだし、その条件式にしたがって検索処理を行ない、結果を要求元へ送る。従って、制御プログラムではキューバッファに検索要求を溜める処理だけを行ない、文字列探索プログラムではキューバッファから検索要求を取り出す処理のみを行なうことになる。文字列探索処理が終わらないまま、制御プログラムが次々と検索要求を受け付けた場合には、キューバッファに待ちキューがどんどん積み上げられていくことになる。

【0042】続いて文字列探索処理の説明をする。この文字列探索プログラムは前述の制御プログラムとは時分割（タイムシェアリング）で動作しているものとする。すなわち、制御プログラムは端末から検索要求が来るのを監視し、文字列探索プログラムは制御プログラムがキューを積むのを待っている。キューバッファに検索要求が入ると文字列探索プログラムは、検索処理を始めるが、検索処理の実行中はキューバッファは参照しないので、次々と検索要求が来たときには検索要求がバッファに溜りつづけることになる。この検索要求は文字列探索プログラムの検索処理の実行終了後、処理されることになる。

【0043】図12に、この文字列探索プログラムの処理の流れを表すPAD図を示す。文字列探索プログラムは、最初検索要求が入ってくるのを待っている。もし、検索要求が入ってくれば検索処理を実行する。この時、待ちキューバッファに入っている検索要求の個数をチェックし、もし複数の検索要求が入っている場合には、これらの検索要求を同時に取り出して一括して検索処理を行う。次にキューの中に入っている条件式を解析し、検索タームにタームIDを振る処理を行う。以後の処理ではこのタームIDを用いて行うことになる。すなわち、文字列探索処理において、テキスト中に該当する検索タームがあった場合、上記タームIDを出力する。そして、ターム間論理条件判定処理で、検索ターム間の論理的な条件すなわち、AND、ORの条件に合致するものだけを出力する。最後に今までの結果を要求端末毎に分配して、送出して一連の検索処理が終了し、再び検索要求待ちの状態へ移る。

【0044】以上の文字列探索処理プログラムの中で、タームIDの付与処理、論理条件の判定処理、結果の分配処理について更に詳細に説明する。

【0045】タームIDの付与処理とは、図13に示すように、単一あるいは複数のキューに蓄えられている条件式の検索タームにユニークな番号を付与することが主な処理となる。その他、条件式間で同じ検索タームが使用されていないか、ターム間のAND条件は存在するか、各検索タームはどの検索端末からの要求によるものかを識別し、処理する。タームID付与処理では、図13に示すように端末識別子と条件式を格納した検索要求を処理して、タームIDテーブル、多重照合タームID

テーブル、AND条件判定テーブル、及び出力IDリストの4個のテーブルあるいはリストを生成する。タームIDテーブルは、実際の文字列と対応するタームIDを示したものである。このとき、ターム間にAND条件があれば、AND条件の項に後述するAND条件判定テーブルの条件IDを登録する。この条件IDは、1から始まる整数値である。

【0046】タームIDテーブルに登録する処理は、検索要求ごとに行われ、それぞれの検索要求内で指定される検索文字列にユニークな番号を付与していく。図13の例では、キュー1の検索条件により“計算機”にタームIDの1が“事務処理”にタームIDの2がつけられることになる。以下、検索タームの登録ごとに順番にユニークなID番号がつけられていく。

【0047】このようにタームIDテーブルにIDをつけていくとき、既に他の条件式で同じ文字列をタームとして用いていた場合、例えば図13の例でキュー3の処理の場合には、多重照合タームIDという特殊なIDをタームIDテーブルに登録する。図13の例では、1001から始まる整数値となっている。そして、多重照合タームIDテーブルに2件分のタームIDを登録しておく。すなわち、図13の例で具体的に説明すると、タームIDテーブルに元々登録されていた“計算機”をあらわすID番号1を1001で置き換え、更に多重照合タームIDテーブルに2件分のタームIDである1と6を登録する。この多重照合タームIDを付与された文字列が文字列検索中ヒットした場合には、多重照合タームIDテーブルを参照してそこに登録された複数のタームIDを検索結果として出力する。図13の例では、“計算機”という文字列が文書中に有ると、タームID1001が出力されるがこれは多重照合タームIDなので、多重照合タームIDテーブルを参照して、結局タームIDの1と6を出力することになる。

【0048】AND条件判定テーブルは、ターム間にAND条件が有ったときに登録される。例えば、図13のキュー1の検索要求を処理するとき、タームIDテーブルにタームIDとターム文字列を登録し、かつAND条件の項目に条件IDの1を登録する。この条件IDも各AND条件ごとにユニークな1から始まる整数値である。また、AND条件テーブルには、条件IDとタームID及び出力IDを登録する。出力IDは、タームIDを含めたユニークな番号となる。図13の例でキュー1の処理をしていて、これまでタームIDは、1と2が付与されているので、AND条件テーブルの出力IDには3が登録されることになる。また、ここでID番号3が使用されるため、キュー2の処理では検索ターム“環境問題”にタームIDの4が付与されることになる。

【0049】AND条件判定テーブルは、図に示すとおりAND条件を持つタームIDとそのタームが文書中にあったことを示すヒットフラグ、及び条件が成立したと

きに出力する出力IDとを項目としている。ヒットフラグは最初0に初期化されており、文字列検索処理において、タームが文書中にあったときに1となる。一つの条件IDに対して全てのヒットフラグが1の場合に出力IDを出力する。また、文書単位でヒットフラグは0にリセットされる。図13の例では、“計算機”（多重照合により、タームIDは1及び6）と“事務処理”（タームIDは2）の両方の文字列が文書中にあったときに条件IDが1の全てのヒットフラグが1になるのでIDの3を出力する。

【0050】出力IDリストは、最終結果を出力するときのタームIDまたは条件式IDを登録したもので、ここに登録されなかったIDは検索結果として出力しない。また、同時に出力するIDと該タームあるいは条件を要求した端末の識別子の対応をとって、照合結果分配プログラムにおいては、この対応をもとに検索結果を端末毎の出力バッファに振り分ける処理を行う。

【0051】以上のタームID付与処理のアルゴリズムをPAD図を用いて詳細に説明する。図14に示すように、この処理は待ちキューバッファより取り出した検索要求の個数分同一の処理を繰り返す。最初に検索要求を出した端末識別子をキューより取得する。次に、条件式の種別に従ってそれぞれの処理を行う。条件式の種別とは、ターム間がどのような論理条件で結ばれているかを区別するもので、本実施例ではAND、OR及び検索タームが単一で用いられていることを示す単純条件の3種類を用いている。AND条件のときは、条件式中のそれぞれのタームについて後述のタームID登録、タームIDテーブルへのAND条件IDの設定、及びAND条件テーブルの設定を行なった後、出力IDリストに端末識別子と出力IDを登録する。OR条件のときは、それぞれのタームについてタームID登録をした後に、各タームIDを出力IDリストに端末識別子とともに登録する。単純条件のときは、タームID登録と出力IDリストへの登録を行なう。

【0052】次にタームIDの登録アルゴリズムの説明をする。この処理は検索タームの文字列をタームIDテーブルに登録するのが主な処理であるが、この時登録しようとする文字列が既に登録されているかチェックし、登録されている場合には多重照合タームIDテーブルへの設定を行なう。すなわち、図15に示すように、まず登録しようとする検索タームの文字列と同一の文字列がタームIDテーブルに登録されているかチェックする。もし、新規であれば新しいタームIDとともに文字列をタームIDテーブルに登録する。既に同じ文字列が存在する場合には、既存タームのタームIDを多重照合タームIDに変更し、多重照合タームIDテーブルに新規タームとともにタームIDを設定する。このとき、既登録のタームにAND条件が設定されていた場合、タームIDテーブルに設定されていたAND条件IDを0にクリ

アし、多重照合タームIDテーブルの方へ設定する。図13の例では、最初“計算機”はタームIDテーブルにAND条件IDとともに登録されていたが、キュー3の処理のとき、多重照合タームID1001に変更し、この時同時にAND条件IDも多重照合タームIDテーブルの方へ設定し直している。

【0053】最後に検索結果の分配処理について詳細に説明する。図16は、本実施例で用いるテキストデータの形態と論理条件判定後のデータの形態及びセッションn用出力バッファに格納されるデータの形態を示している。テキストデータは、文書を特定するための文書IDのdxxxを先頭として複数文書が列になって格納されている。図16の例では、d001からd004までのテキストデータを文字列判定処理、論理条件判定処理を行なって文字列の有った文書番号とヒットしたタームIDを出力する。この時は、各要求元から来た条件は全て一緒に出力される。この論理条件判定処理後のデータを図13に示した出力IDリストをもとに各要求元に対応するセッションn用出力バッファへ結果の文書IDを振り分ける。すなわち、論理条件判定の出力するIDの2と3の出現した文書はセッション3用出力バッファへ、ID5はセッション5用出力バッファへ、ID6はセッション1用出力バッファへそれぞれ文書IDを出力するればよい。

【0054】以上、CPU110の処理について詳細に説明した。次に、磁気ディスク120に複数のデータベースが格納されている時の処理について説明する。データベースは図17に示すように、磁気ディスク120上にデータベース毎にまとまって格納されているものとする。

【0055】この様に複数個のデータベースを格納して、各要求元が対象とするデータベースを指定して検索要求を出す場合、制御プログラムがキューバッファに溜った検索要求を一括して処理しても処理時間の短縮につながらない場合がある。つまり、各要求が別々のデータベースに対するものであった時である。例えば、要求元1がデータベース1を検索対象に指定し、要求元2がデータベース2を検索対象に指定した検索要求がキューバッファに溜っていた時、これらの検索要求を統一して、データベース1と2の両方を一括して探索しても、別々に探索してもトータルのスキャン量は変わらない。すなわち、図17の例では磁気ディスク上の論理アドレス0から3500番地までをスキャンする量は変わらないのである。

【0056】これに対し、要求元1と2の両方が同じデータベース2を対象に指定しているときには、検索要求を統一して一括して探索するとスキャンする量が半分になることがわかる。すなわち、検索要求を一つずつ別々に処理すると、論理アドレス2100番地から3500

番地を2度スキャンするのに対して、統一して処理すれば1度のスキャンで探索処理が終了できることがわかる。

【0057】この様に、情報検索装置が複数のデータベースを格納する場合には、各要求の検索対象データベースを考慮し、探索範囲を同じくする要求にたいして、検索要求を一括処理することによって、スキャン量を減らし、待ち時間の少ない処理を実現することができる。

【0058】このために、検索装置側では各検索端末から送られてくる検索要求をキューバッファに格納する際に、どのデータベースが対象かを識別するためのデータベースIDも同時にキューの中に格納する。この時のキューバッファの構造は、図18のようになる。すなわち、端末識別子、データベースID、条件式をキューとして格納することになる。そして、このキューバッファから、溜っている検索要求を取り出して統合処理するとき、データベースIDが同一の検索要求のみを選択することで、前述の効果的な一括検索処理が可能となる。

【0059】このデータベース選択の検索処理を具体的に説明する。図19は図12で説明した文字列探索プログラムの複数データベースを有するときの処理を示すPAD図である。処理の違いはキューバッファからのキューの取り出しで、それ以外の処理は既に説明したので省略する。今、図18の様にキューバッファに検索要求が溜っている場合、まずキュー1をバッファから取り出して、データベースIDを参照し、バッファ内のデータベースIDが2であるキュー、すなわちキュー1と対象が同一であるものだけをキューバッファから取り出す。この例では、キュー3とキュー4が該当するため、キュー2だけをバッファに残し、残りの検索要求は統合され一括処理される。この様に、検索要求を統合し、データベース2を一度スキャンした後、端末識別子1, 2, 5にそれぞれ検索結果を分配し検索処理を終える。この時、キューバッファにはまだキュー2が残っているの

で、次にこの検索要求がバッファから取り出され、データベース3について検索処理を始めることになる。

【0060】この様に、複数のデータベースに対して要求が送られてくる場合でも、複数の検索要求を一括して処理することが可能で、待ち時間の少ない情報検索装置を提供することができる。

【0061】以上、単一あるいは複数の検索要求が有る場合の文字列検索処理について説明した。このようにして得られた検索結果である文書の集合は、図20のようにして格納され、以後の検索の際にベース集合として用いたり、あるいは集合間のOR演算をしたりして用いられる。図20の例では、検索結果集合である文書IDの列を文書IDの位置を1とし、ヒットしていない文書IDの位置を0とした検索結果ビットリストで表現している。ビットリストで表現した理由は、集合間のAND、OR処理を行なうのにビット間のAND、OR演算で処

理できるという利点があるからである。このビットリストを検索回数分蓄えて管理しているのが、結果集合管理テーブルである。この結果集合管理テーブルは、接続している端末の個数分用意され、それぞれに各端末の要求毎の結果集合が格納されている。以下、この結果集合管理テーブルを用いたハイアラーキ検索の処理について説明する。

【0062】ハイアラーキ検索とは、図21に示すように、結果集合管理テーブルで管理する過去の結果集合を検索範囲として、検索する処理のことで、絞り込み検索とも呼ぶ。すなわち、過去（通常直前の検索結果）の検索結果である文書IDの列を結果集合管理テーブルのビットリストから求め（これをベースと呼ぶ）、新たな条件式（図21の例では“学習型ユーザインタフェース”）でテキストデータを探索する。この時、テキストデータのベース以外のデータは読み飛ばす処理を行なう。例えば、図21の例ではベース以外のテキストデータd001～d002及びd004～d014などは不要テキストとして、文字列の探索処理を行なわない。これが絞り込み処理である。結果として、ベースのテキストデータに条件の検索タームが存在した文書ID d008, d016... が出力される。複数の検索要求が待ちキューバッファに溜っている場合にこのハイアラーキ検索を実現する方法を次に説明する。検索要求のあった端末が異なる場合、当然ベースも異なることになる。従ってベースをどの様に設定するか、また結果集合をどのように分配するかが問題となってくる。文字列の照合動作は、テキストデータをスキャンするという正確上できるだけ少なくすることが望まれる。そこで、図22のように、各要求元の結果集合管理テーブルからベースのビットリストを読みだし、それらのビットORを複数要求受付時のベースとする。そして、絞り込んだ後の結果集合をまたビットリストに戻し、各要求元のベースビットリストとAND演算する。最後のAND演算は、要求元のベース集合以外の他の要求元のベースでの条件式の照合によるノイズを取り去るために必要となる。

【0063】以上ハイアラーキ検索機能の実現処理を含め、第一の実施例の説明をした。本実施例は、特別なハードウェアを必要とせず、単一のCPUで複数のフルテキストサーチ要求を処理するのに効果がある。

【0064】本実施例では、各端末からユーザがオンラインで検索要求を発信することを仮定に説明したが、オフラインで指定された検索要求に対しても同様の処理で複数の検索処理を一括して処理することができる。これは、例えばバッチジョブのように何件もの検索要求を一台の検索装置で処理する場合に、従来は一件ずつDBのテキストデータをスキャンして結果を出力していたものが、一回のスキャンですべての要求を処理できることを意味する。つまり、本実施例によればオンライン、オフラインを問わず高速な検索要求の処理が可能となる。

【0065】これより、第二の実施例の説明を行なう。本実施例は、文字列探索処理のための特別なハードウェアあるいは探索処理専用のCPUを有する複数のCPUで実現することに特徴がある。

【0066】図23は第二の実施例の構成を示す図である。本実施例は、検索端末1300、検索処理を実行するCPU1310、制御プログラムと各種のデータバッファを格納するメモリ1320、CPU1310に接続し実際の文字列探索処理を制御するためのCPU1330、文字列照合処理を専門に処理する文字列照合部1340、データベースを格納する磁気ディスクの書き込み読出しを制御するディスク制御部1350、磁気ディスク1360、文字列照合処理を制御するプログラムと各種データエリアを格納するメモリ1370とからなっている。1330～1370はTSMという一つの処理ブロックを形成してCPU1310に複数個接続する。メモリ1320には、TSM及び端末を制御する制御プログラムが格納されている。また、同メモリは、各TSMが持つデータベースを管理するDB-TSM対応テーブル、各TSM用の待ちキューバッファ、接続された端末用のセッションn用出力バッファとしても使っている。メモリ1370には、文字列照合処理を制御する制御プログラム、ターム間のAND、OR条件を判定する論理条件判定プログラム、複数の端末からの検索要求を同時に受け付けたときに検索結果集合を分配する照合結果分配プログラムが格納してある。また、同メモリは照合結果を一時蓄える照合結果出力バッファ、各接続端末毎に結果集合間利テーブルの格納領域としても用いる。これより、検索処理の流れに従って、各プログラムの処理の詳細を説明する。

【0067】端末側での検索処理の流れは、第一の実施例と同一である。第一の実施例と異なる点は、CPU1310、1330の制御プログラムの流れであるので、これらについて詳しく説明する。図24はCPU1310の処理の流れを示すPAD図である。基本的にCPU1310は端末からの要求を待っているが、割込みによりTSMからの検索処理終了信号も処理する。まず、端末からの要求の処理について説明し、次にTSMからの終了信号の処理について説明する。

【0068】端末から接続要求があった場合には、要求の検索対象となるデータベースからDB-TSM対応テーブルを参照し、対応するTSMへ接続要求を発行する。その後、セッションn用出力バッファの領域をメモリ1320に確保する。解放要求の場合は、TSMへ解放要求を出し、接続時に確保したセッションn用出力バッファの領域を解放する。検索要求の場合は、対応TSMの待ちキューバッファへ端末識別子と共に検索要求を登録する。文書データ表示要求の場合は、対応TSMへデータ表示要求を出し、データが出力バッファに格納されるのを待って要求元端末へデータを送出する。TSM

からの検索処理終了割込みを受け付けた場合、検索要求元の端末が指定したデータベースを担当する全てのTSMの処理が終わったかを判定し、全てのTSMの検索処理が終了している場合にかぎり出力バッファからの検索データの送出行う。それ以外の場合は、終了割込みがあっても、これを無視して全てのTSMの処理が終了するまで待っている。

【0069】ここで用いるDB-TSM対応テーブルは図25のような構造をしている。項目として、DB名称とそのDBを格納しているTSMの番号の対応表となっていて、接続要求時、端末からの検索対象データベース名称からこのテーブルを参照して、対応するTSMへ接続要求を出す。例えば、新聞記事を検索対象とする端末からの接続要求が来た場合には、TSM1と2に接続要求を出し、以後該端末からの検索要求はTSM1と2に対して行われる。

【0070】キューバッファの構造は第一の実施例と同様であるが、TSMの数だけメモリ1320に用意されており、端末からの検索要求にしたがって対応するTSM用のキューバッファに端末識別子と共に登録される。例えば、新聞記事DBを検索対象とする端末からの検索要求では、TSM1と2用のキューバッファにそれぞれ検索要求が登録される。

【0071】次にCPU1330の処理について説明する。このTSMに設けられたCPUでは、上位のCPU1310からの要求を待っている。図26に示すように、接続の要求があったときには、接続された端末用の結果集合管理テーブルの領域を確保する。また、開放要求の場合には、対応する端末用の結果集合管理テーブルの領域を開放する。文書データ表示の要求では、送られて来た文書IDに対応するテキストデータをディスク制御部を駆動して、磁気ディスクより取り出し、上位CPU1310の管理するセッションn用出力バッファへデータを送出する。この時の、送り先であるセッションn用出力バッファとは、勿論データ表示要求元の端末に対応する出力バッファを示す。

【0072】検索要求の場合は、第一の実施例と同様に、まず待ちキューバッファに蓄えられた検索要求の数をチェックし、1個なら単一キュー取り出し、複数個ならばキューを同時に取り出して、タームID付与処理を行う。この後、文字列照合部とディスク制御部を駆動し、条件式中の検索タームを探索する文字列探索を行う。そして、ターム間の論理条件判定を行って条件式に合致する文書IDを照合結果出力バッファに出力した後、端末識別子を基に照合結果を各要求元別に分配して、上位CPUの管理するセッションn用出力バッファへ出力する。この時、結果集合は要求元に対応する結果集合管理テーブルへも格納し、ハイラークサーチ等での検索のために利用する。最後に、上位CPU1310へ担当分の検索処理が終わったことを報告して、全て

の検索処理が終わり、また上位CPUからの要求を待つ。

【0073】以上、第二の実施例を説明した。本実施例によれば、文字列探索処理のための特別なハードウェアあるいは探索処理専門のCPUを有するため、複数のTSMで並列にフルテキストサーチを実行することができる。

【0074】また、この第二の実施例では、処理ブロックTSMが複数個CPUに接続し、同CPUが検索端末のあるLANに接続する構成としたが、図27に示すようにTSMを検索端末と同一のLANに接続する構成でも同じ処理を行うことが可能である。

【0075】さらにまた、検索端末からの接続要求により確保されるセッションn用結果集合管理テーブルとセッションn用出力バッファを、同一端末からの別々の接続要求で個々に確保することで、同一端末上の異なる検索セッションをそれぞれ管理することもできる。

【0076】

【発明の効果】本発明によれば、CPU占有時間の多いフルテキストサーチ処理を複数の端末をつなげて処理することができる。すなわち、複数端末から同時に検索処理要求が来たときでも、後から要求のあった端末を待たせることなく、一回のテキストスキャンで複数個の条件式を一括で処理することが可能となる。

【0077】待ちキューバッファを設け、端末から来た要求を逐次登録しておくことにより、検索処理中に入ってきた要求は、現在の検索処理終了後に、複数の要求をまとめて処理することができるようになる。

【0078】まとめて処理された検索結果は、照合結果分配プログラムと、セッションn用出力バッファにより要求元別に分配され、それぞれの要求元へ送出することができる。

【0079】オフラインで複数件の検索要求を処理する場合においても、同様の処理を行うことで、複数の検索要求を一括して処理することができるので、一件単位で検索要求を処理するよりも効率的に検索要求を消化することができるようになる。

【0080】また、要求元別に結果集合を蓄える結果集合管理テーブルを持ち、ハイアラキーサーチのときに、第一に要求元分のベース集合のOR集合を検索対象として検索処理し、第二に得られた照合結果を要求元別に分配し、第三に分配した結果集合に対して各要求元のベース集合とANDすることで、複数の要求元から来る検索要求を一括して処理することが可能となる。

【0081】さらに、端末からの要求を受け付ける上位CPUとそれにつながる複数個の文字列照合を担当する下位CPUとを有することにより、テキストスキャン処理を複数台で並列に行うことができ、高速なフルテキストサーチが可能となる。

【図面の簡単な説明】

【図1】キューバッファを用いた複数要求受け付け処理を示す概念図である。

【図2】複数の検索要求の一括処理を示す概念図である。

【図3】マルチユーザでのハイアラキー検索の処理を示す説明図である。

【図4】マルチユーザでのハイアラキー検索の効率的な処理を示す説明図である。

【図5】一括検索処理において検索ノイズが発生することを示す説明図である。

【図6】一括検索後の分配処理を示す概念図である。

【図7】一括検索後の分配処理が正しい結果を出力することを示す集合関係図である。

【図8】第一の実施例の構成図である。

【図9】本実施例における検索端末側の検索業務の流れを示すPAD図である。

【図10】検索装置の制御プログラムの流れを示すPAD図である。

【図11】待ちキューバッファの構造を示す概念図である。

【図12】文字列探索プログラムの処理を示すPAD図である。

【図13】タームID付与処理を示す概念図である。

【図14】タームID付与処理を示すPAD図である。

【図15】タームIDの登録処理を示すPAD図である。

【図16】検索結果の分配処理を示す概念図である。

【図17】データベースの格納形式を示す概念図である。

【図18】待ちキューバッファの構造を示す概念図である。

【図19】複数のデータベースを有するときの文字列探索プログラムの処理を示すPAD図である。

【図20】検索結果格納方法を示す概念図である。

【図21】ハイアラキー検索の原理を示す概念図である。

【図22】複数要求受け付け時のハイアラキー検索の方法を示す概念図である。

【図23】第二の実施例の構成図である。

【図24】上位CPUの制御プログラムの処理を示すPAD図である。

【図25】DB-TSM対応テーブルの形態を示す図である。

【図26】下位CPUの制御プログラムの処理を示すPAD図である。

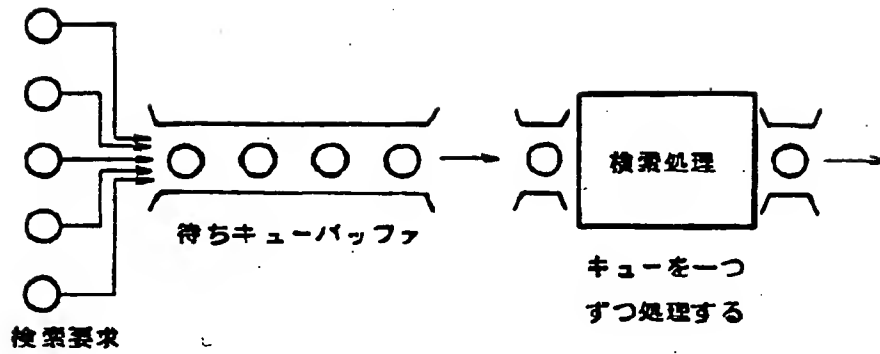
【図27】検索端末と検索処理ブロックを同一のLANで接続した構成図である。

【符号の説明】

100…検索端末、110…CPU、120…磁気ディスク、130…メモリ。

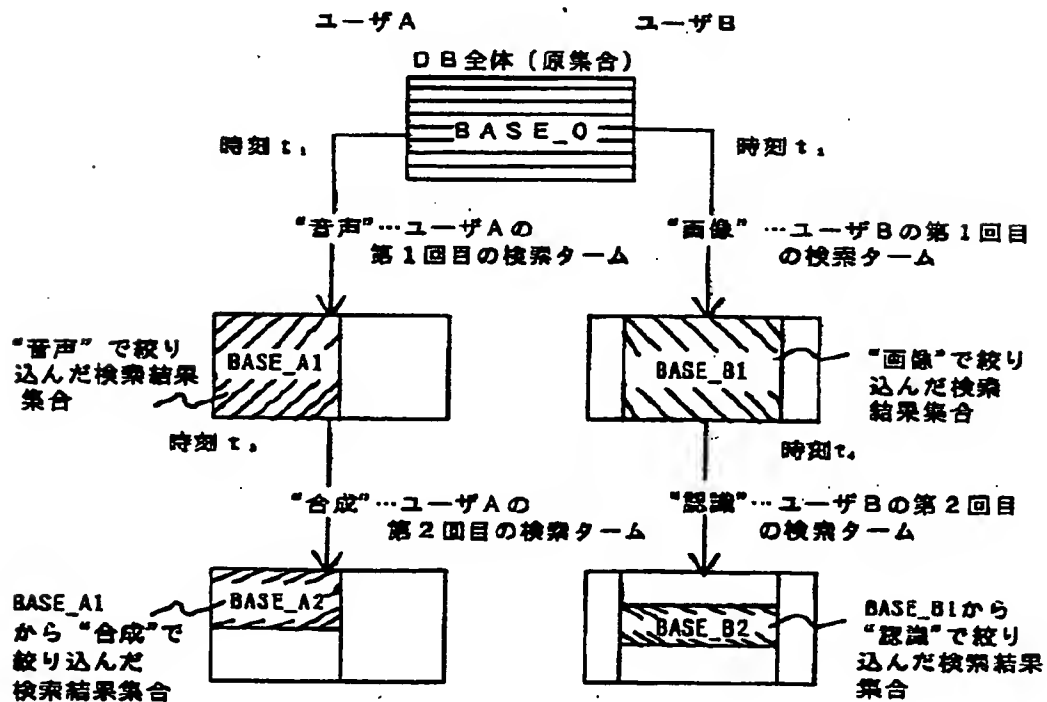
【図1】

図 1



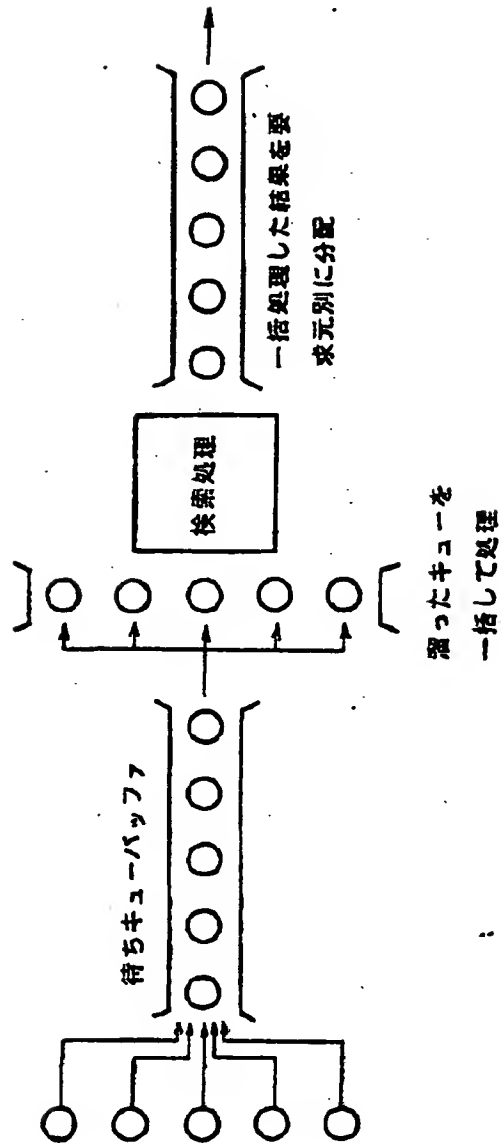
【図3】

図 3



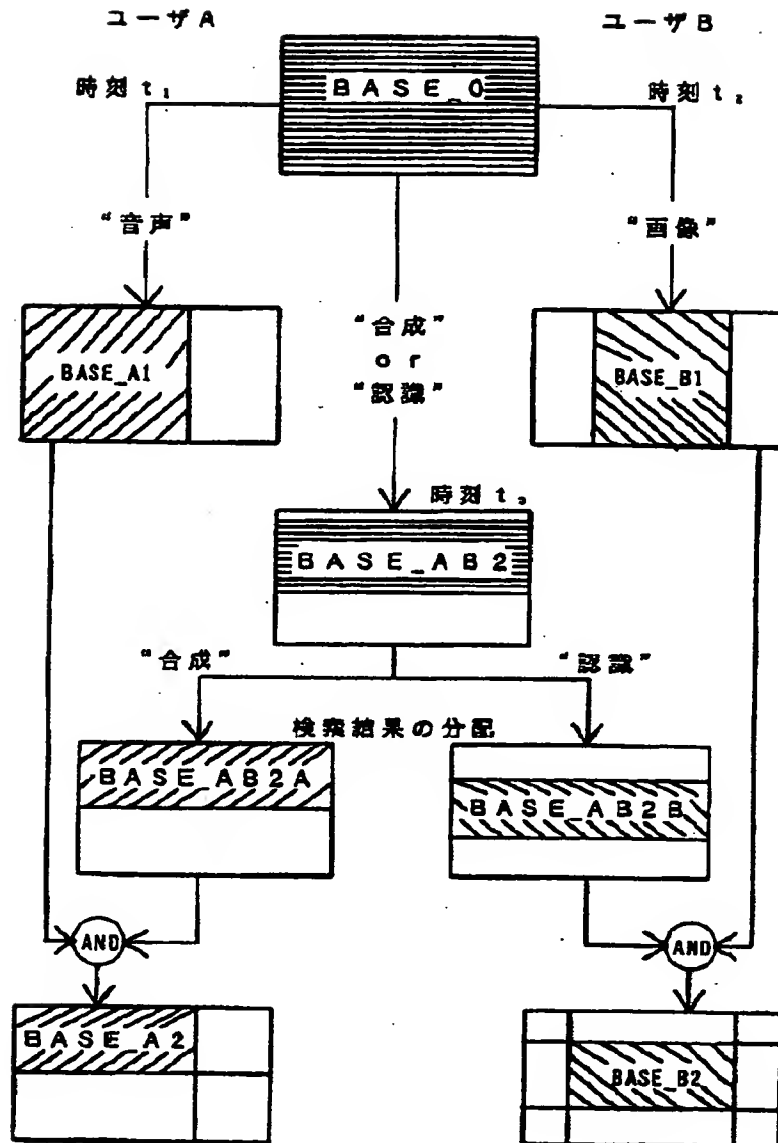
【図2】

図 2



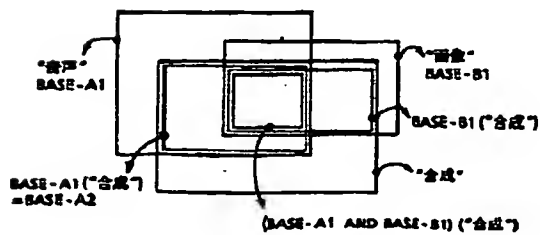
【図4】

図 4



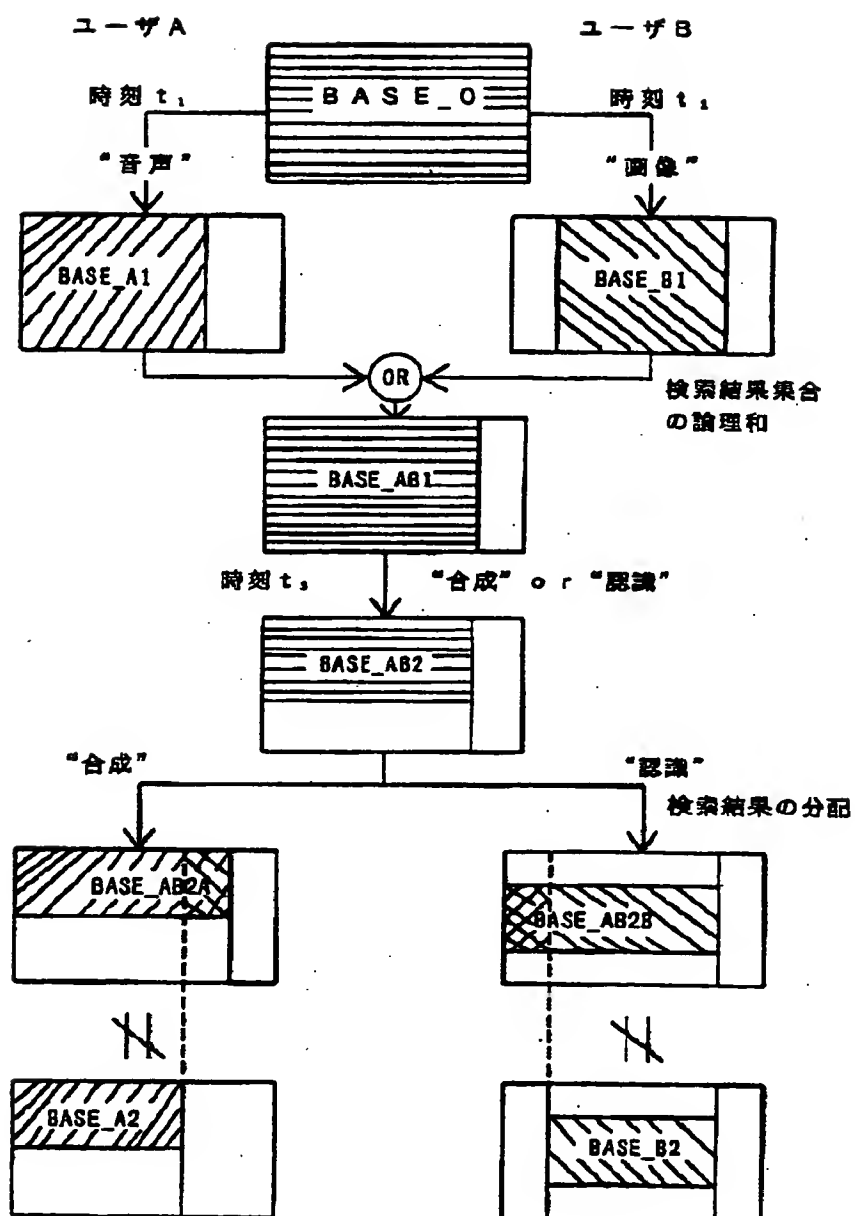
【図7】

図 7



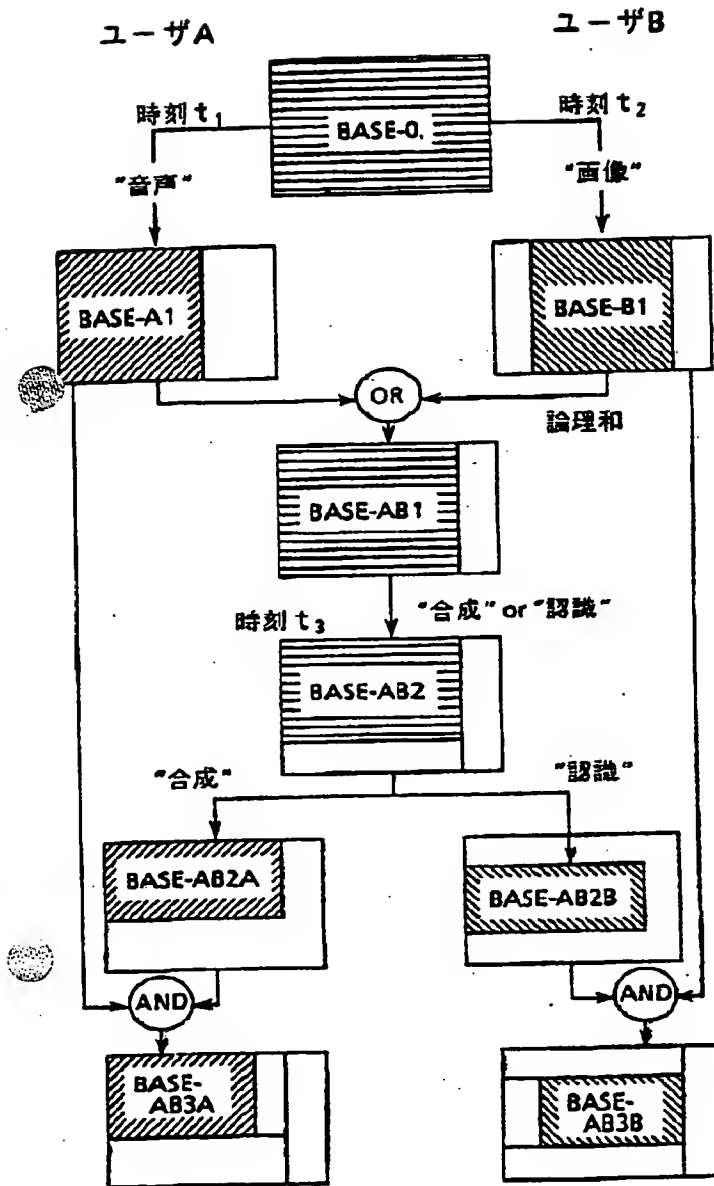
【図5】

図 5



【図6】

図6



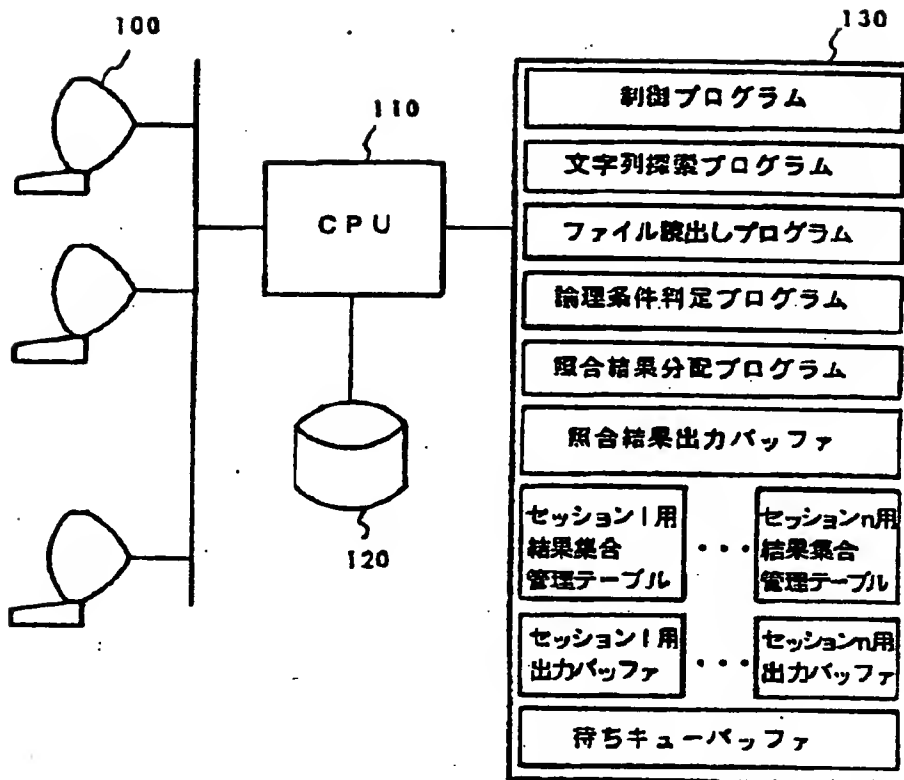
【図25】

図 25

DB名	対応TSM番号
新聞記事	1. 2
議事録	3
マニュアル	4
報告書	5. 6
論文	7. 8. 9

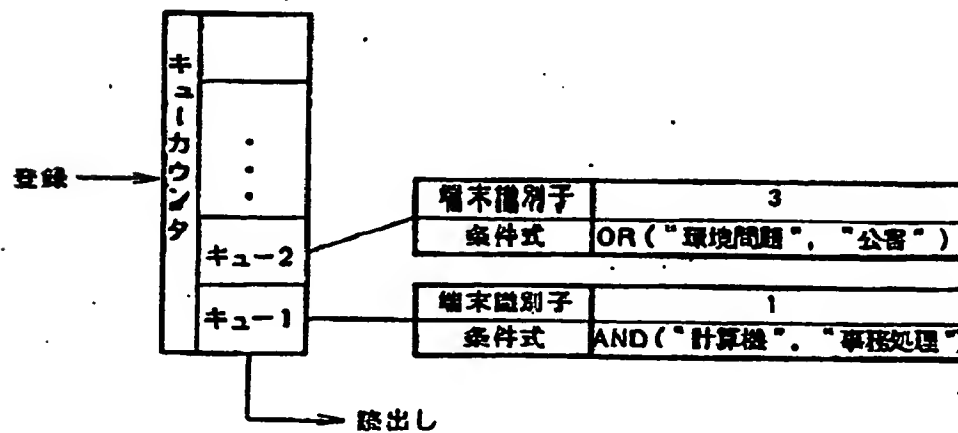
【図8】

図 8



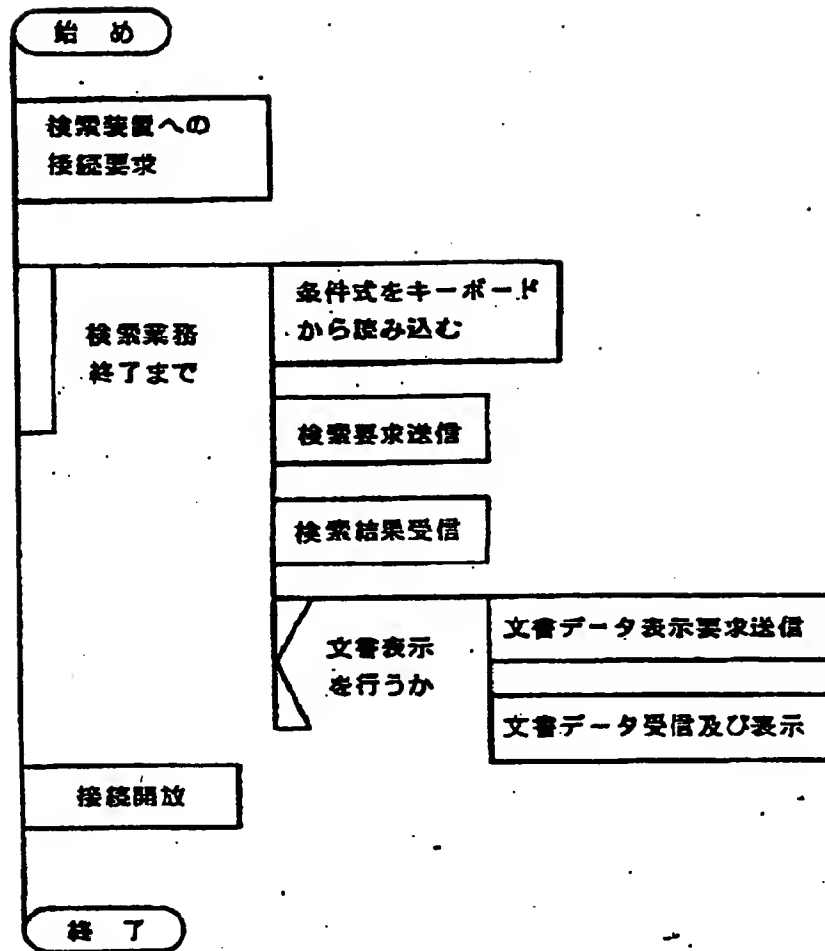
【図11】

図 11



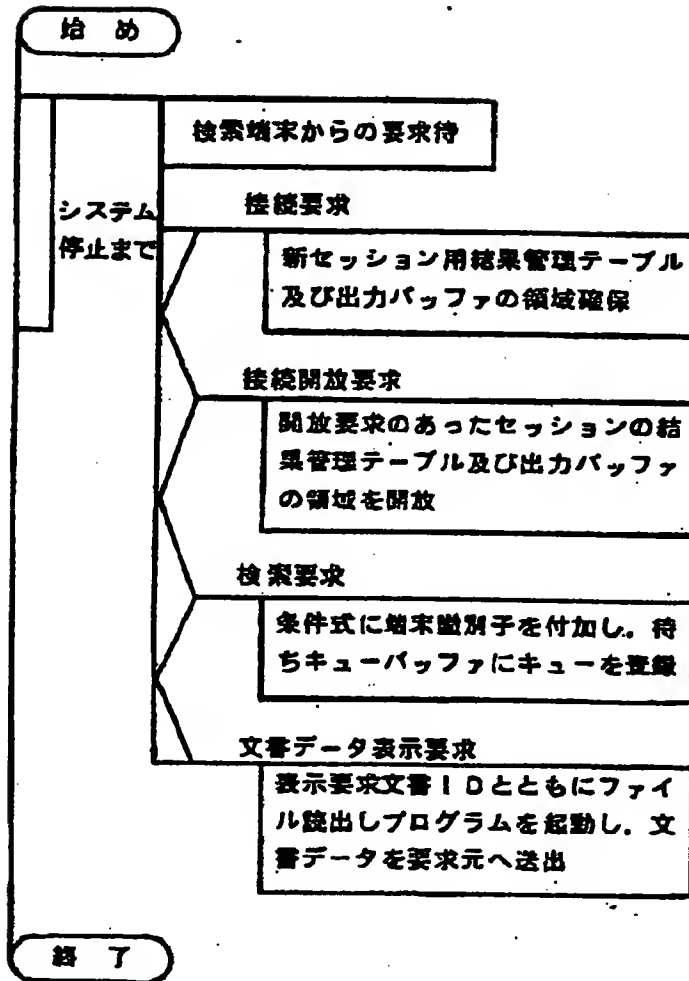
【図9】

図 9



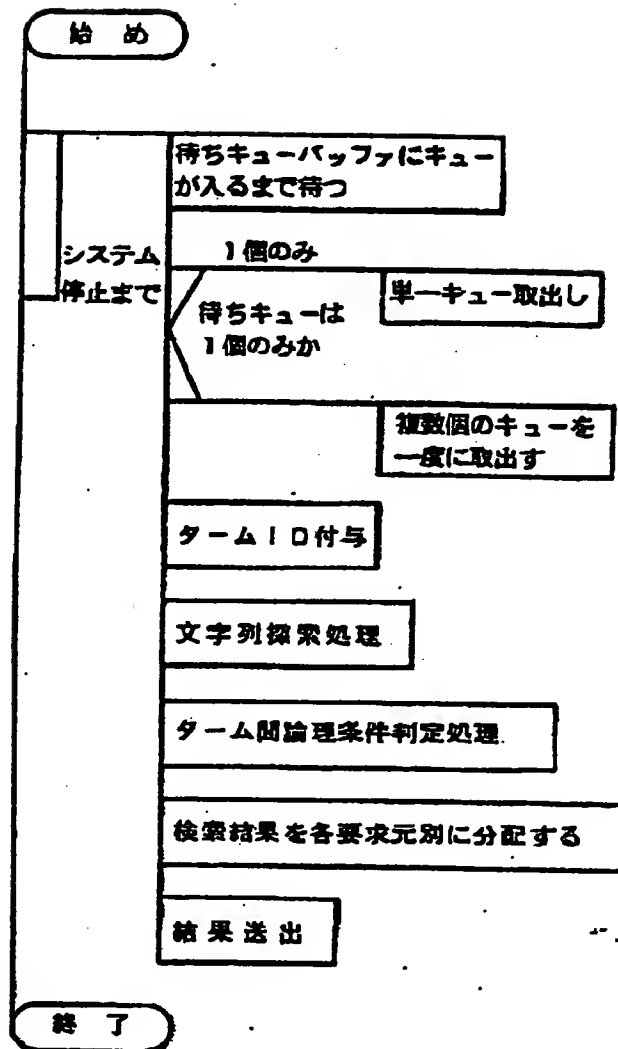
【図10】

図 10



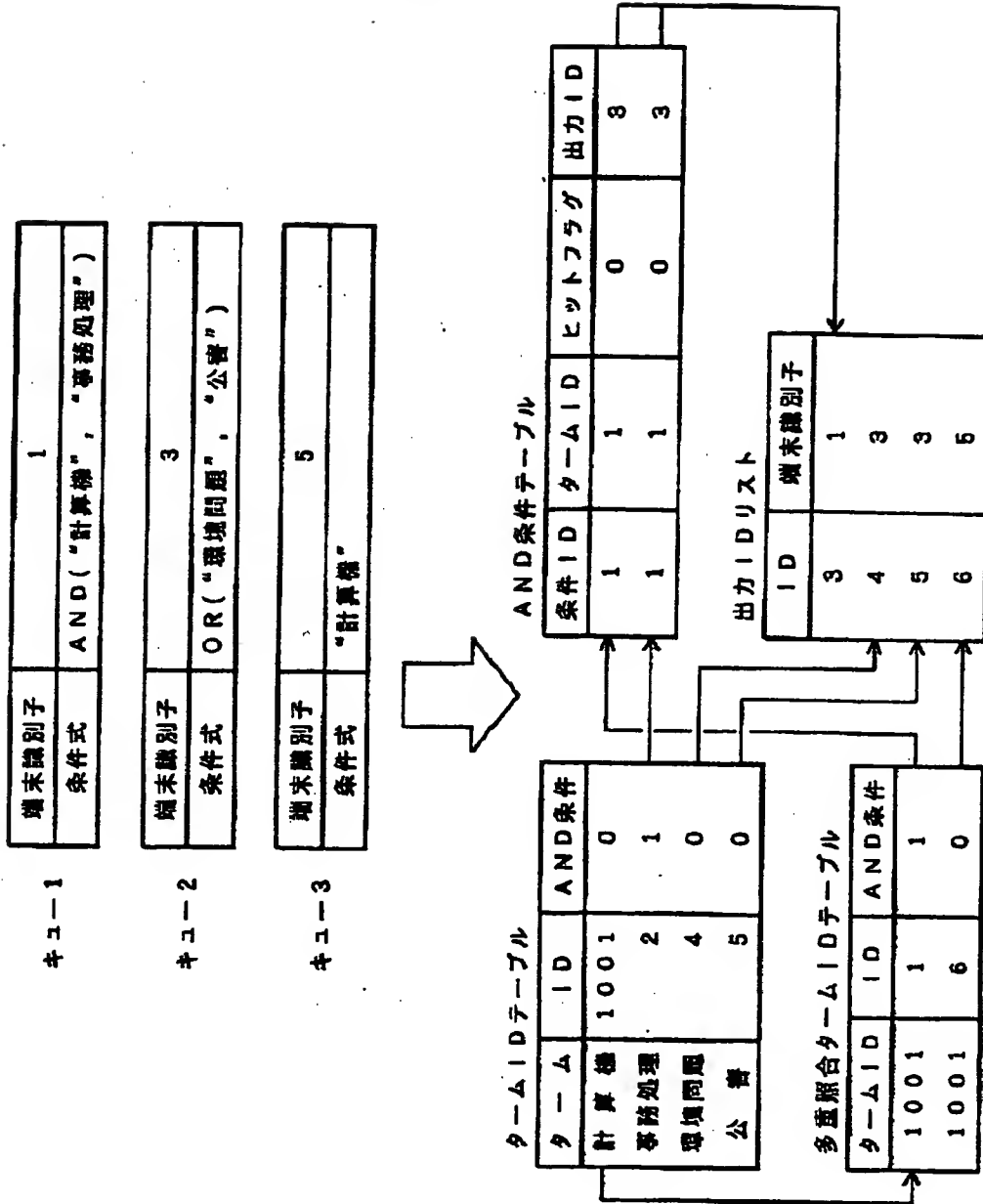
【図12】

図 1 2



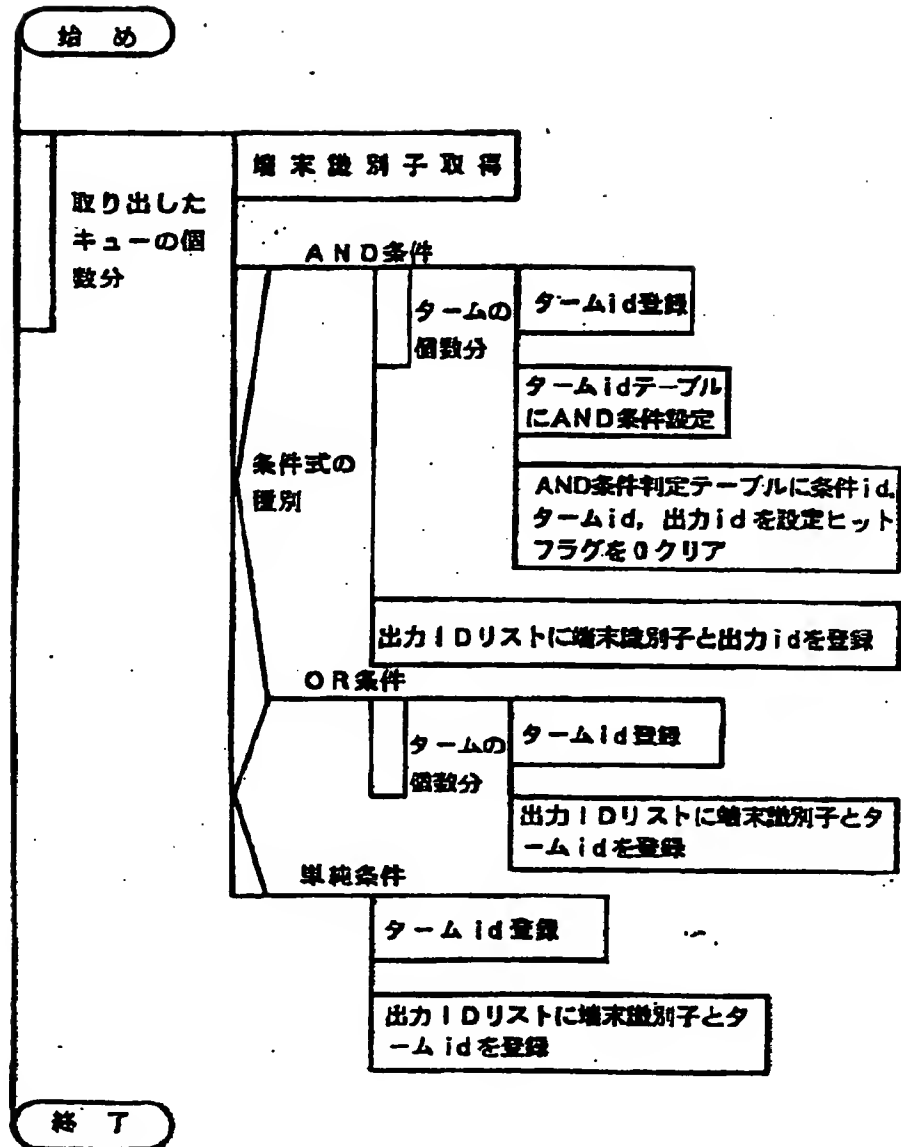
【図13】

図13



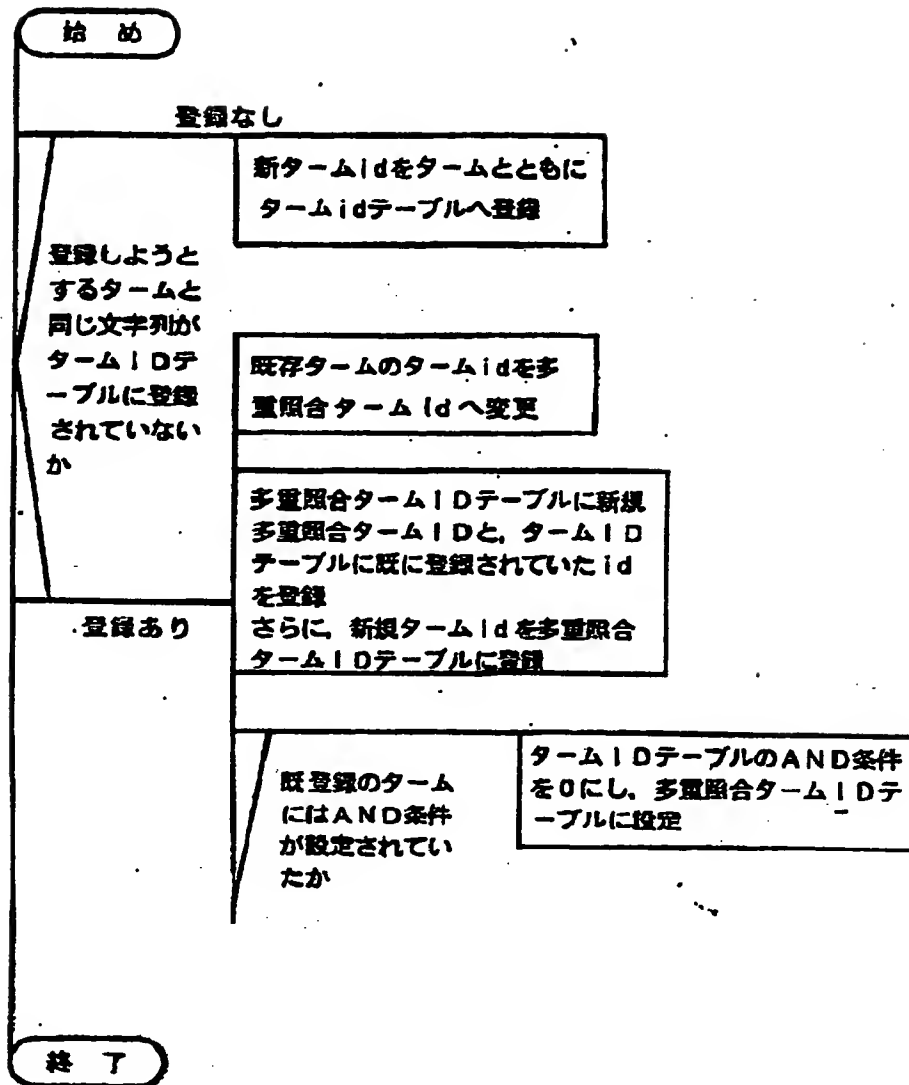
【図14】

図 14



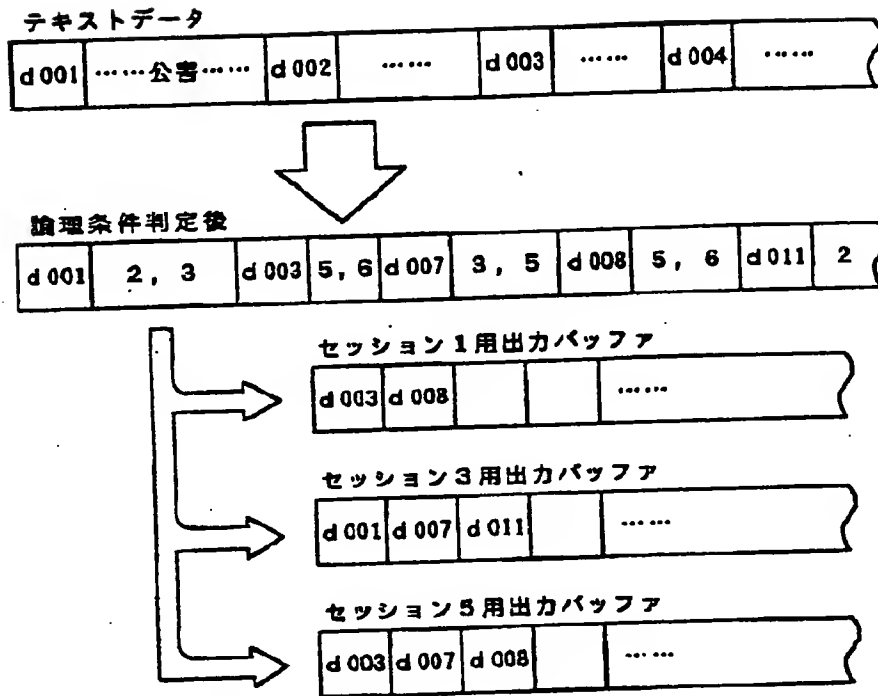
【図15】

図 1 5



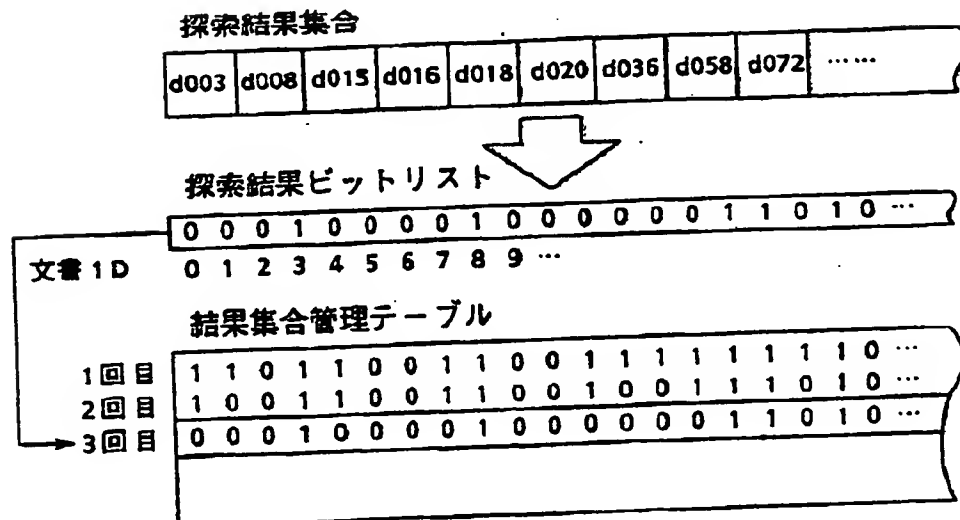
【図16】

図 1 6



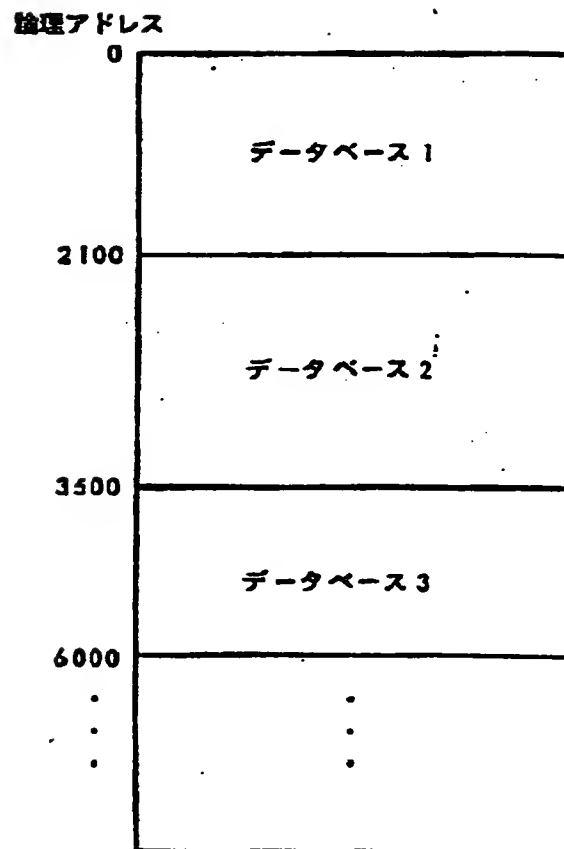
【図20】

図 20



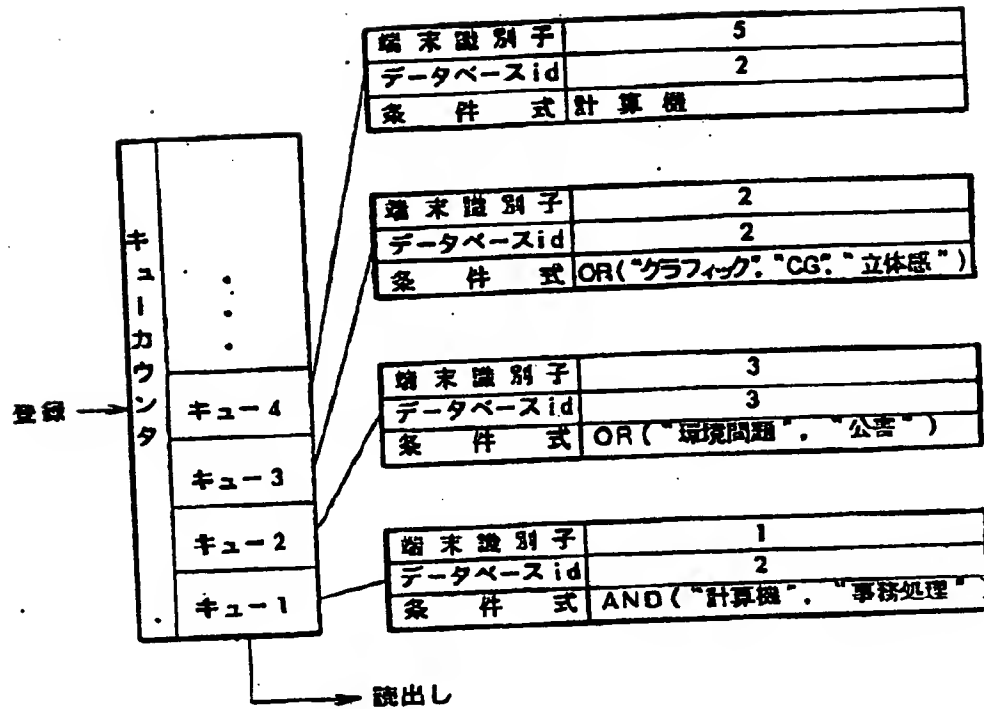
【図17】

図 17



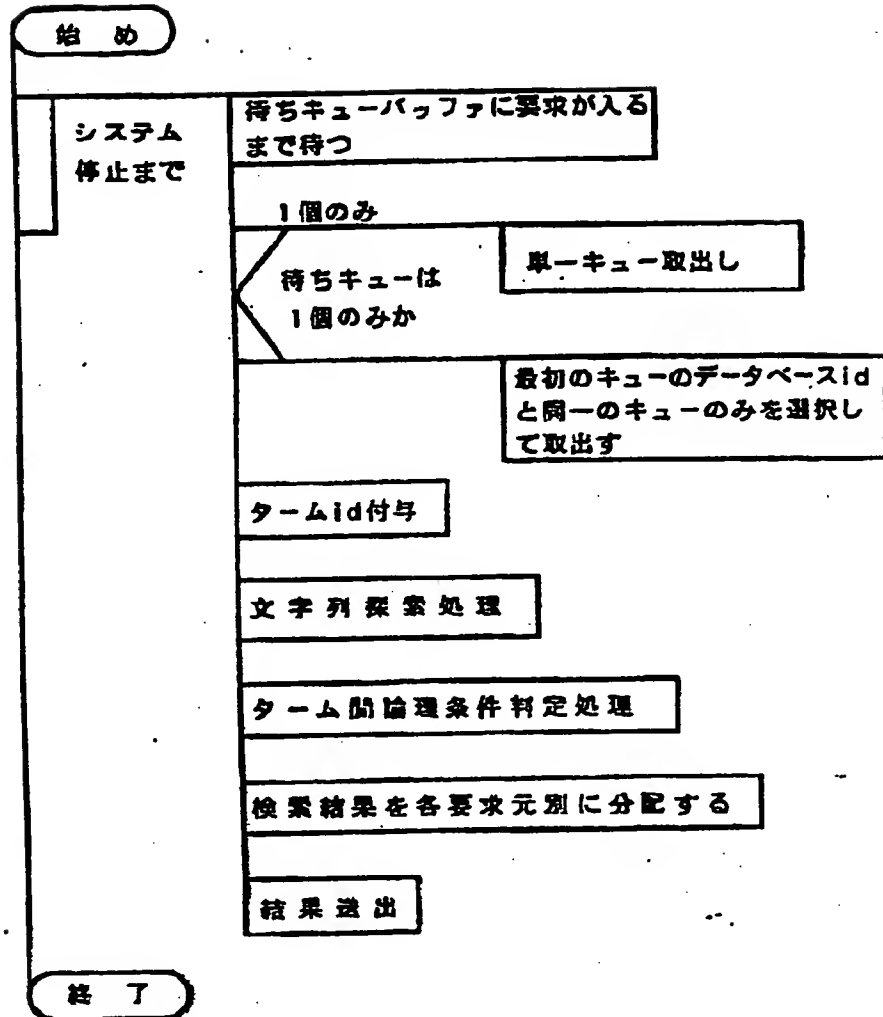
【図18】

図 18



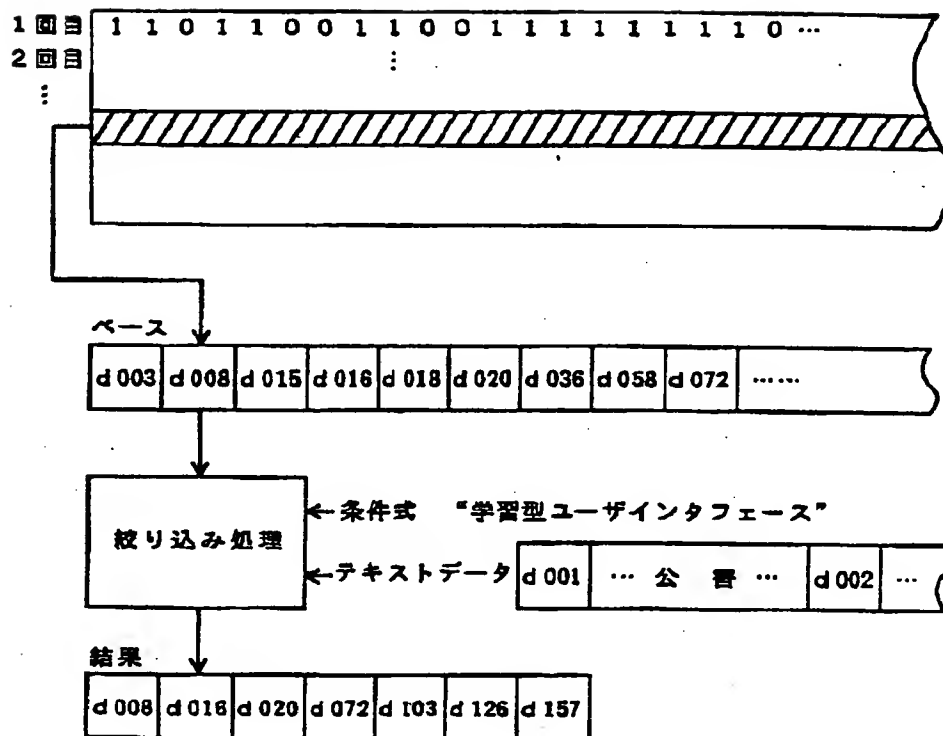
【図19】

図 19



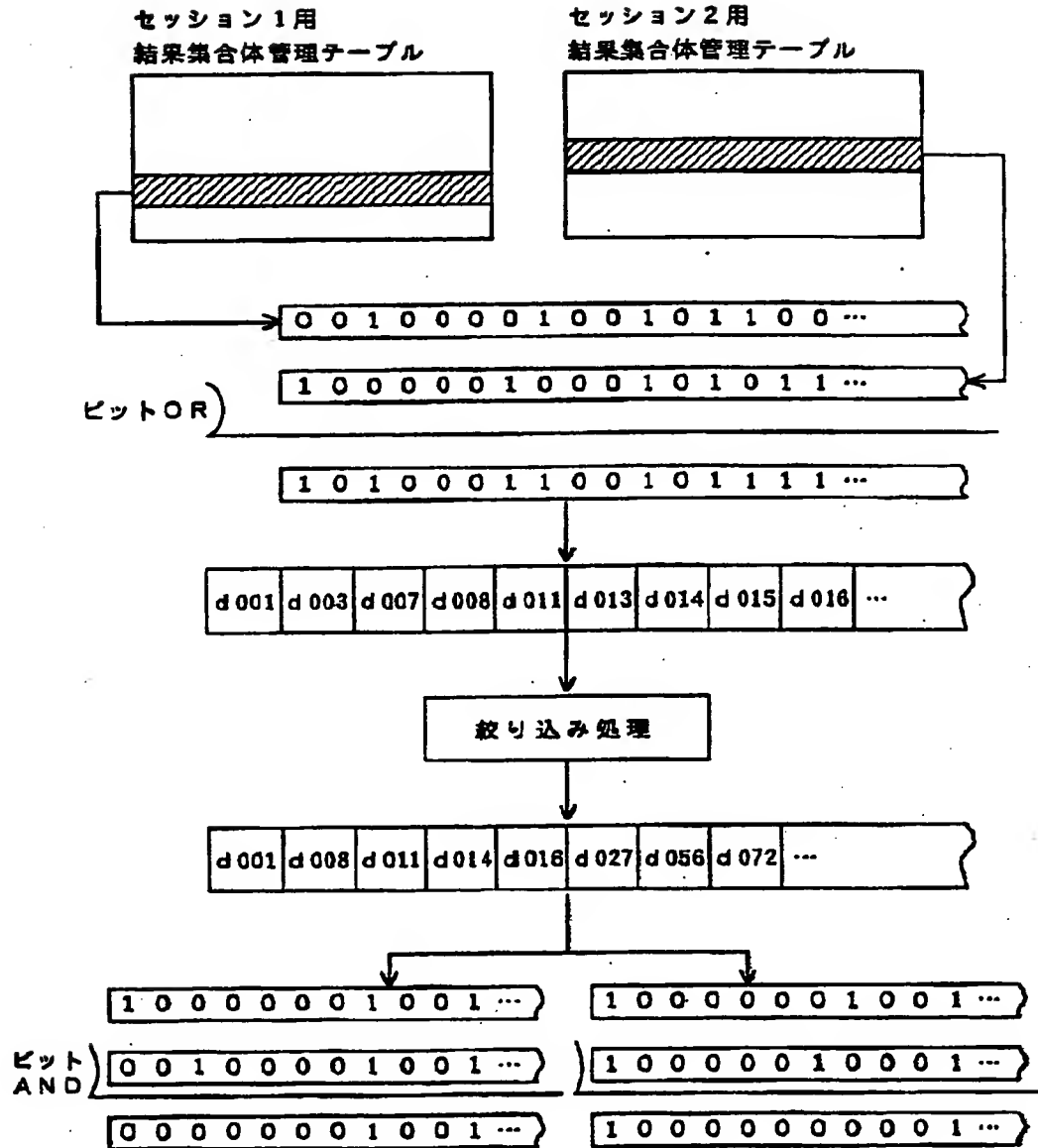
【図21】

図21



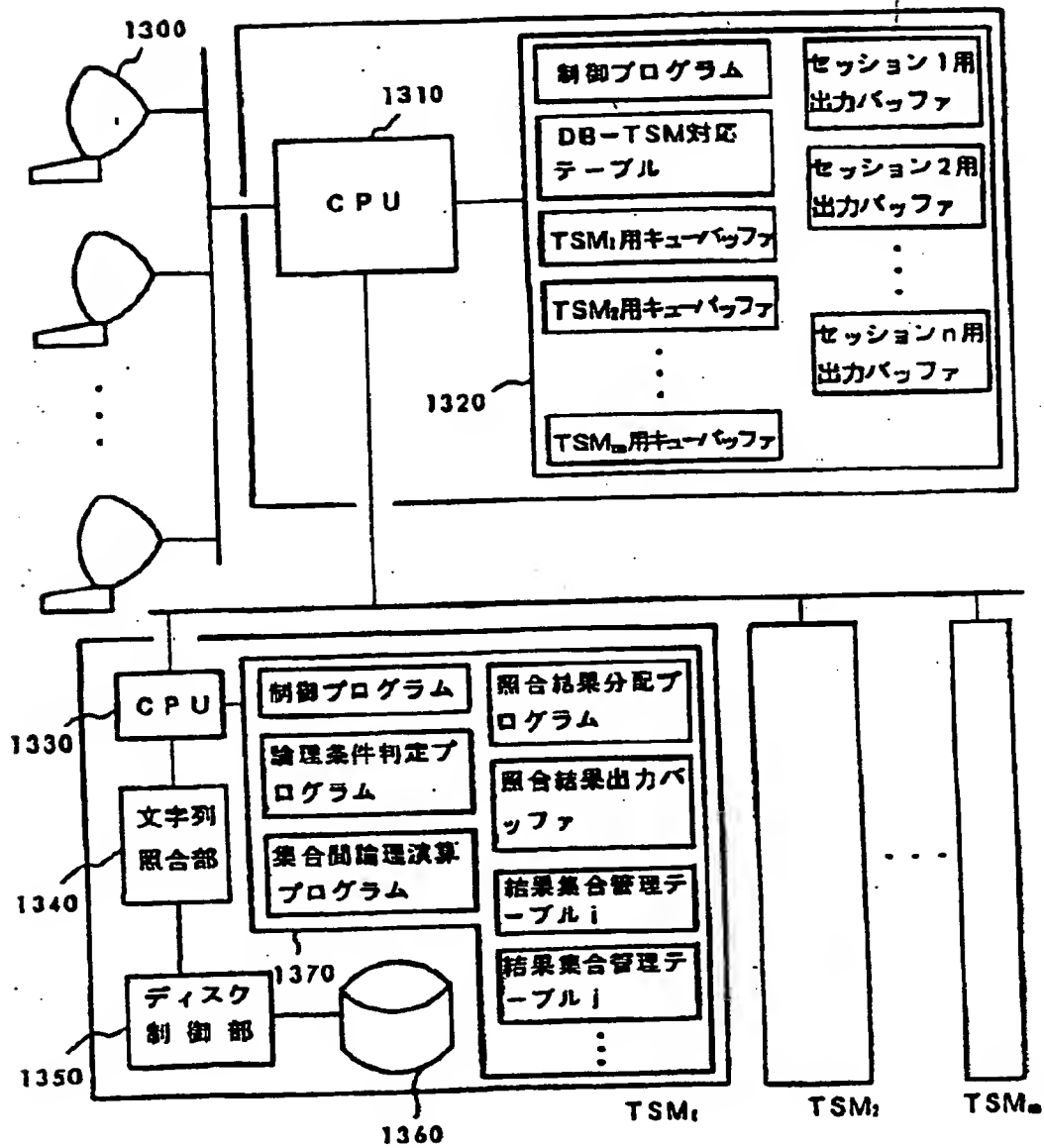
【図22】

図 2 2



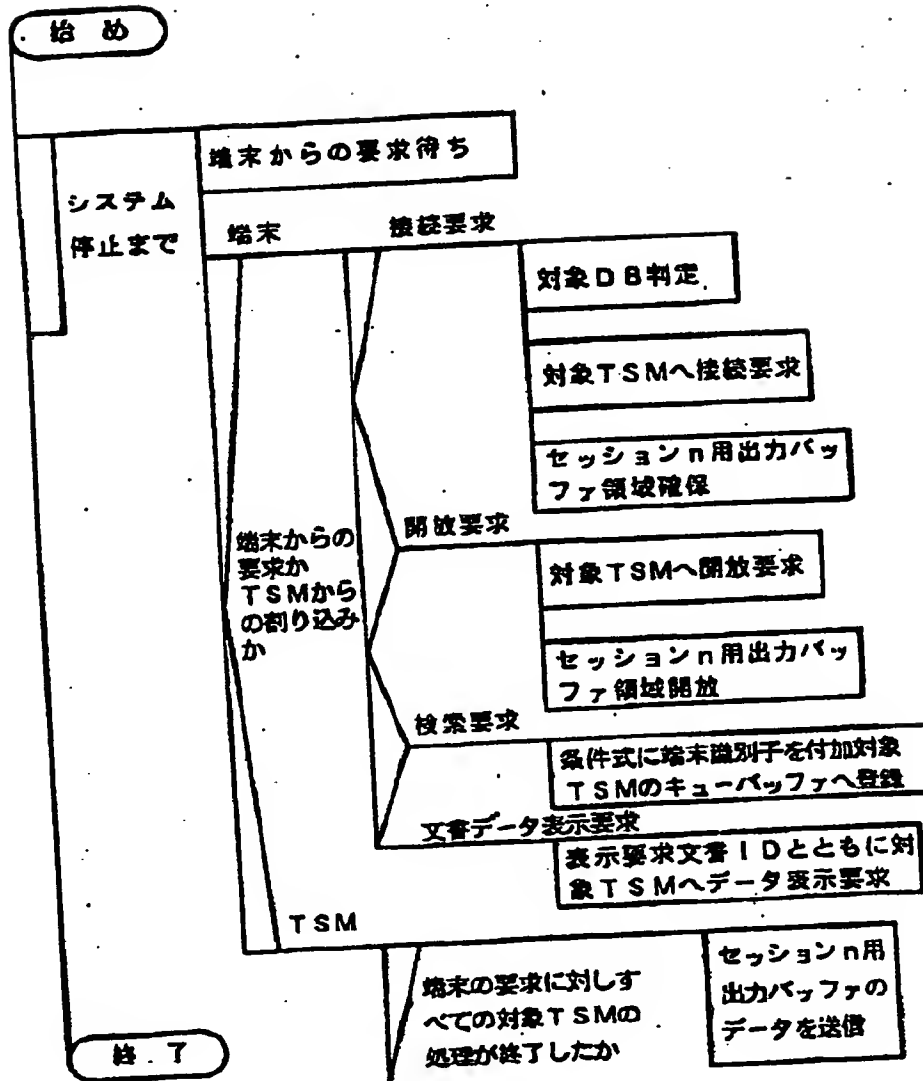
【図23】

図 23



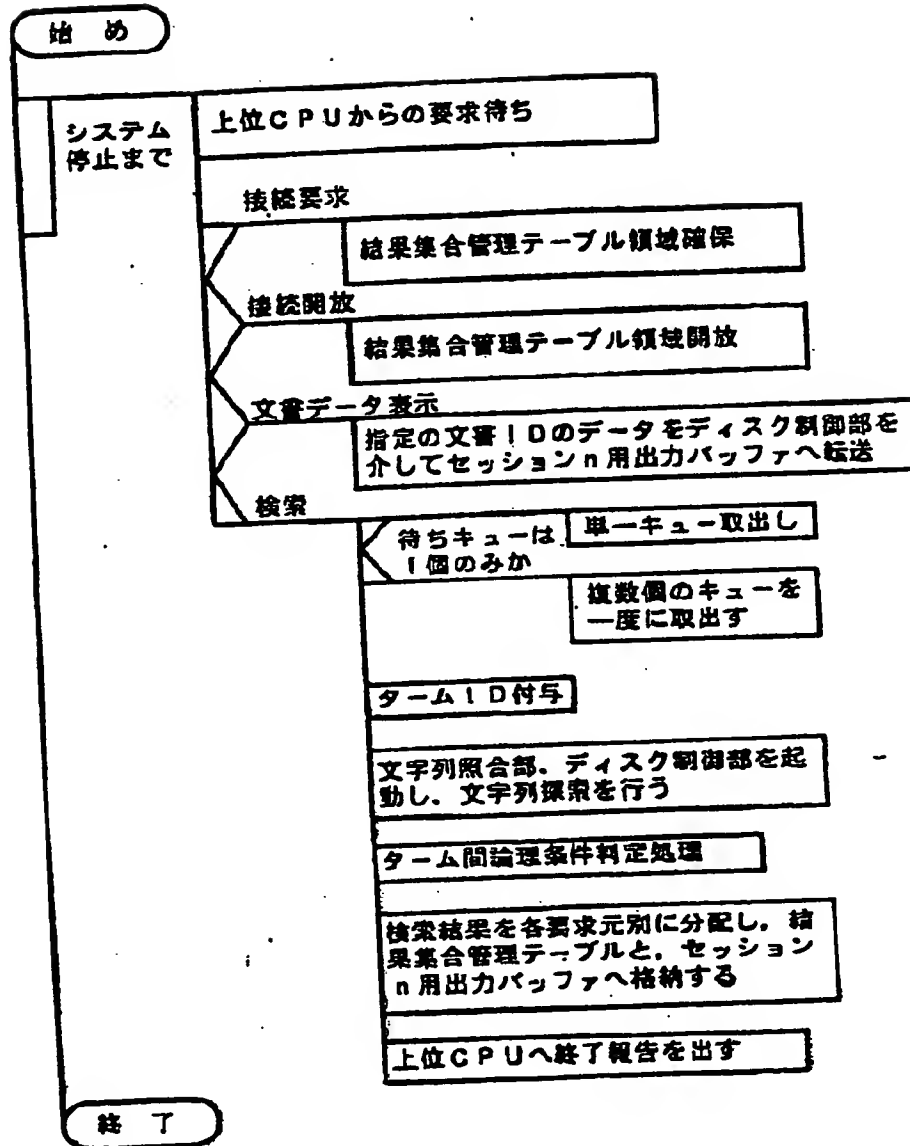
【図24】

図 24



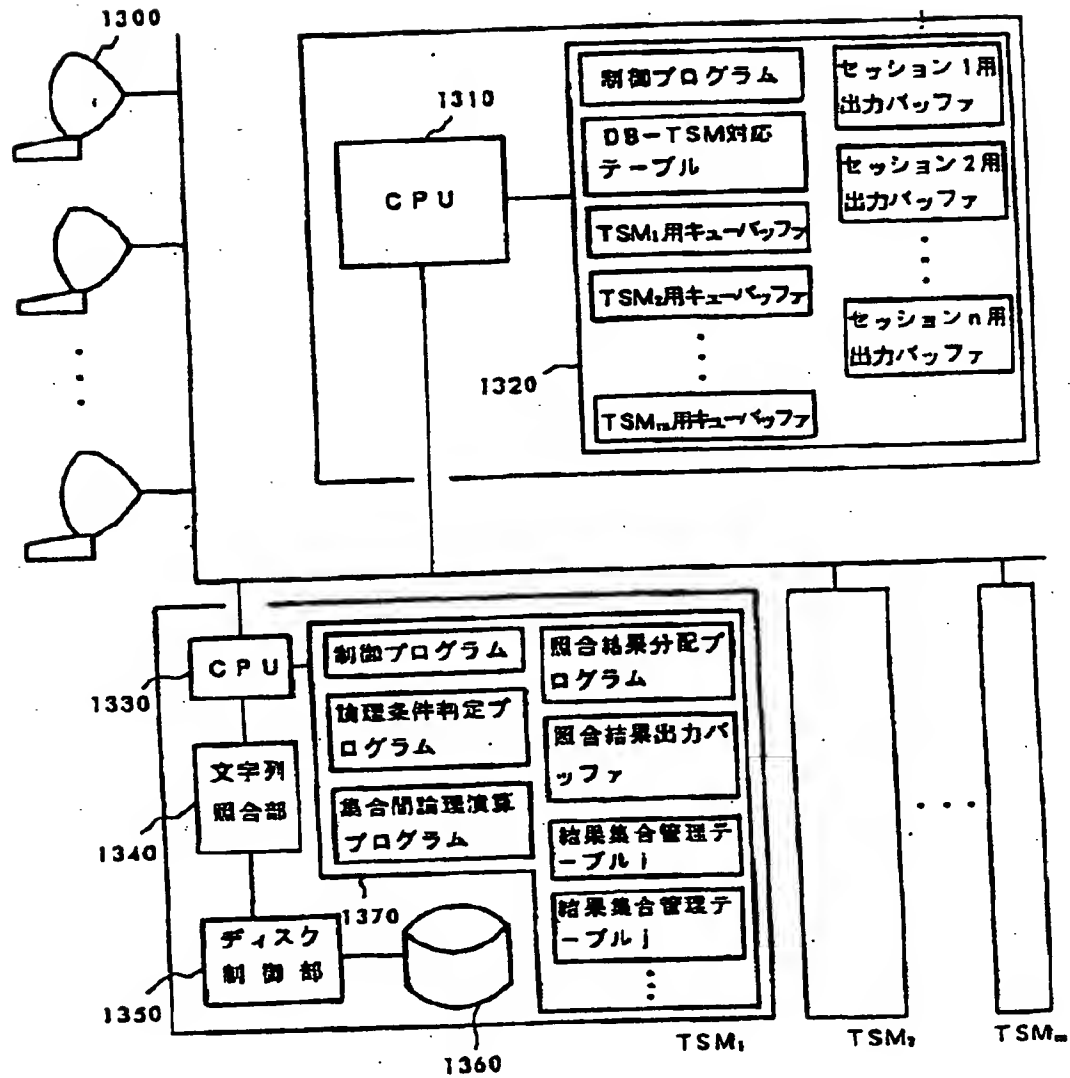
【図26】

図 26



【図27】

図 2 7



フロントページの続き

(72) 発明者 川口 久光

東京都国分寺市東恋ヶ窪1丁目280番地
株式会社日立製作所中央研究所内